

## Método Multiobjetivo de Aprendizaje para Razonamiento Inductivo Difuso

Jesús Acosta\* Angela Nebot\*\* Josep M. Fuertes\*

\* Dpt. d'Enginyeria de Sistemes, Automàtica i Informàtica Industrial  
Universitat Politècnica de Catalunya,  
Edif. U (FME), Campus Sud. C/ Pau Gargallo 5. 08028 Barcelona  
Teléfono: (34) 93-4016974; Fax: (34) 93-4017045  
[jesus.acosta@upc.edu](mailto:jesus.acosta@upc.edu), [josep.m.fuertes@upc.edu](mailto:josep.m.fuertes@upc.edu)

\*\* Dpt. de Llenguatges i Sistemes Informàtics  
Universitat Politècnica de Catalunya,  
Edif. Omega-Campus Nord. C/Jordi Girona 1-3. 08034 Barcelona  
Teléfono: (34) 93-4137783; Fax: (34) 93-4137833  
[angela@lsi.upc.edu](mailto:angela@lsi.upc.edu)

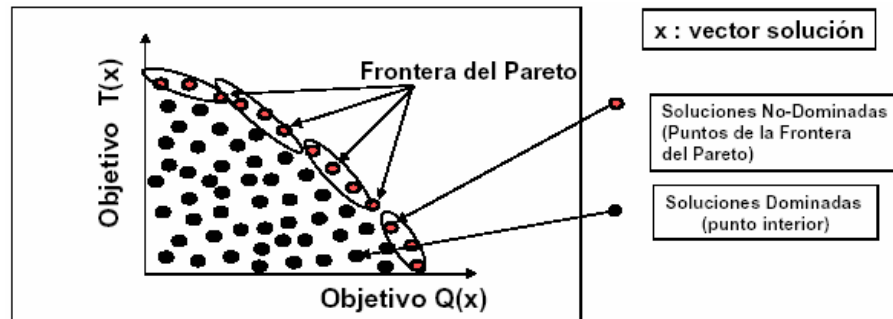
**Resumen:** Se ha reconocido en varios estudios que las variaciones en la granularidad (número de clases por variable) y en las funciones de pertenencia tienen un efecto significativo en el comportamiento de los sistemas difusos. La metodología FIR no es una excepción. La eficiencia de los procesos de identificación del modelado cualitativo y de predicción de FIR está muy influenciada por los parámetros de discretización de las variables del sistema, es decir, del número de clases de cada variable y de las funciones de pertenencia que definen su semántica. Para resolver ésta problemática se han planteado en trabajos previos varias metodologías híbridas denominadas *Sistemas Genéticos Difusos* (SGDs) que intentan aprender de manera conjunta o por separado dichos parámetros. Estos métodos han usado funciones monobjetivo para la evaluación de los cromosomas. En esta investigación se presenta otro método de aprendizaje automático que permite obtener a la vez los parámetros de fusificación de la Metodología FIR (número de clases y funciones de pertenencia) pero ahora utilizando Algoritmos Genéticos Multiobjetivo (AGMOs). Se describen en detalle los componentes principales del nuevo método desarrollado y se presentan los resultados obtenidos sobre una aplicación medioambiental.

**Palabras clave:** Algoritmos genéticos multiobjetivo, razonamiento inductivo difuso, sistemas genéticos difusos, machine learning, concentraciones de ozono, contaminación del aire, modelado medioambiental.

## 1. Acerca de los algoritmos genéticos multiobjetivo

La mayoría de los problemas de optimización del mundo real [BSEK06] tienen un nivel de complejidad elevado debido a que poseen varias funciones objetivo, si bien éstas tienden a simplificarse transformándolas (todas excepto una) en restricciones adicionales al problema. De este modo, un problema de optimización multiobjetivo (MOP) se puede definir como el problema de encontrar un vector de variables de decisión que satisfaga ciertas restricciones y optimice un vector de funciones cuyos elementos representan las funciones objetivo [CVL02, Deb01, VL00a]. Estas funciones forman una descripción matemática de criterios de rendimiento que están normalmente en conflicto entre ellos. Por la tanto, el término "optimización" se refiere a la búsqueda de una solución tal que contenga valores aceptables para todas las funciones objetivo.

En general, la optimización multiobjetivo no se restringe a la búsqueda de una única solución, sino de un conjunto de soluciones llamadas *soluciones no-dominadas*. Cada solución de este conjunto se dice que es un *óptimo de Pareto* y, al representarlas en el espacio de los valores de las funciones objetivo, conforman lo que se conoce como *frente o frontera de Pareto* (Véase figura 1). Dado un problema concreto, la obtención del frente de Pareto es la principal finalidad de la optimización multiobjetivo.



**Figura 1.** Frontera del Pareto para  $[Max Q(x), Max T(x)]$ .

Las diversas técnicas existentes para obtener el frente de Pareto se pueden clasificar en 3 categorías: enumerativas, determinísticas y estocásticas [CVL02]. En los últimos años, los métodos estocásticos han sido ampliamente estudiados; en particular, un gran número de autores han trabajado en el área de *Algoritmos Evolutivos (AEs)* [FF95, Coe99, VL00a], sobre todo en *Algoritmos Genéticos (AGs)*. Estos métodos no garantizan la solución óptima, pero ofrecen soluciones aceptables para un amplio rango de problemas de optimización en los que los métodos deterministas encuentran dificultades. La búsqueda enumerativa, que es un método determinista en el que no se emplea heurística alguna, constituye una estrategia conceptualmente simple de optimización, y está basada en la evaluación de cada posible solución dado un espacio de búsqueda finito. El inconveniente de

esta técnica es su inherente ineficiencia, ya que puede ser computacionalmente costosa e incluso prohibitiva a medida que el espacio de búsqueda crece.

El uso de los AEs para resolver problemas de optimización multiobjetivo se plantea por primera vez a finales de los 1960s en la tesis doctoral de Rosenberg [Ros67]. Sin embargo, el primer intento real de atacar un problema multiobjetivo fue el *Vector Evaluated Genetic Algorithm (VEGA)* desarrollado por Schaffer en su tesis doctoral y presentado en la 1<sup>ra</sup> Conferencia Internacional de Algoritmos Genéticos y sus Aplicaciones celebrada en Pittsburgh, 1985 [Sch84, Sch85]. Esta nueva área de investigación denominada *Optimización Multiobjetivo Evolutiva (EMOO)*, ha crecido considerablemente como indica el incremento notable (significativo en los últimos 15 años) de artículos técnicos en conferencias internacionales y revistas, sesiones especiales y grupos de interés en Internet<sup>1</sup>. Todos los algoritmos desarrollados en esta área son llamados *Algoritmos Evolutivos Multiobjetivo (AEMOs)*.

Existen cuatro tipos de AEs bien definidos que han servido como base a la mayoría del trabajo desarrollado en el campo de la IA: los *Algoritmos Genéticos (AGs)*, las *Estrategias Evolutivas (EEs)*, la *Programación Evolutiva (PE)* y la *Programación Genética (PG)*. Sin embargo, en esta investigación se describe una aportación adicional que se basa en el uso de AGs para resolver problemas de optimización multiobjetivo, es decir, *Algoritmos Genéticos Multiobjetivo (AGMOs)*. A continuación se describen algunos aspectos relacionados a la nueva temática tratada que son de gran interés.

En un sentido general, los AGMOs se definen como aquellos AGs que permiten resolver problemas de optimización multiobjetivo, es decir, problemas en donde se deben satisfacer diferentes objetivos. Por ejemplo, un sistema de control de aire acondicionado debería: a) Minimizar el consumo de energía, b) Maximizar el confort y c) Maximizar la estabilidad del sistema de control. Por lo tanto, no existe una única solución sino que es necesario satisfacer varias simultáneamente (que forman la frontera de *Pareto*).

El mecanismo de trabajo de los AGMOs es parecido al de los AGs tradicionales, estos requieren de una evolución de varias poblaciones soluciones al problema y el uso de mecanismos que mantengan la diversidad en la población para conseguir un conjunto de soluciones no dominadas lo más grande posible. Asimismo, se inician con un conjunto de configuraciones aleatorias llamada población inicial. Cada individuo (cromosoma) en la población representa una solución al problema de optimización. En cada generación, los individuos son evaluados usando una función de adaptabilidad (*fitness*). Basados en ese valor

---

<sup>1</sup> C. Coello ha elaborado un portal en Internet sobre Optimización Multiobjetivo Evolutiva, titulado "Evolutionary Multiobjective Optimization (EMOO) Web Page", con muchos enlaces a artículos de revistas, artículos de conferencias, libros, capítulos de libros, reportes técnicos, tesis de master, tesis doctorales, eventos actuales, software de dominio público y principales grupos de trabajo alrededor del mundo. Disponible vía <http://www.lania.mx/~ccoello/EMOO/>

algunos individuos, llamados padres, son seleccionados. La probabilidad de selección de un individuo está relacionada con su adaptabilidad. Se busca asignar una mayor probabilidad de selección a los mejores individuos. Luego un número de operadores genéticos son aplicados a los padres para producir nuevos individuos que formarán parte de la nueva población. El proceso continúa intentando obtener soluciones cada vez mejores hasta que un criterio de parada sea satisfecho.

Pueden considerarse en general, dos tipos principales de AEMOs:

1. Los algoritmos que no incorporan el concepto de óptimo de *Pareto* en el mecanismo de selección del algoritmo evolutivo sino que usan funciones agregativas lineales. Estos implementan un modelo evolutivo utilizando pesos para la agregación de los objetivos. Entre ellos están *Vector Evaluated Genetic Algorithm* (VEGA) [Sch84, Sch85], *Variable Objective Weighting GA* (VOW-GA) [HL92], *Random Weights GA* (RW-GA) [MI95, TIG98], etc.
2. Los algoritmos que jerarquizan a la población de acuerdo a si un individuo es no dominado (usando el concepto de óptimo de *Pareto*), es decir, utilizan un modelo evolutivo que genera poblaciones de soluciones no dominadas. Entre ellos se encuentran *Multiobjective Optimization Genetic Algorithm* (MOGA) [FF93], *Non-dominated Sorting GA* (NSGA) [SD93, SD94], *Niched Pareto GA* (NPGA) [HN93], *NSGA II* [DAPM00, DPAM02], *Multi-Sexual GA* [LE97], *Pareto Archived Evolution Strategy* (PAES) [KC99a], *micro-Genetic Algorithm* (micro-GA) [CT01a, CT01b], *micro-GA2* [TC03], etc.

Históricamente se puede considerar la existencia de dos generaciones de AEMOs:

1. **Primera Generación:** Caracterizada por el uso de jerarquización de *Pareto* y nichos. Algoritmos relativamente simples. También se produjeron enfoques más rudimentarios (por ejemplo, funciones agregativas lineales). Entre los más destacados están VEGA, MOGA, NSGA, NPGA, etc.

Como se ha señalado el precursor del primer AEMO fue David Schaffer [Sch84, Sch85] quien extendió el programa GENESIS de Grefenstette [Gre84] para incluir funciones multiobjetivo. El enfoque de Schaffer utilizó una extensión del *Simple Genetic Algorithm* (SGA) que él llamó *Vector Evaluated Genetic Algorithm* (VEGA) y que difiere del primero en la manera en la cual el mecanismo de selección se lleva a cabo. La selección divide a la población, y en cada sub-población se realiza la selección basada únicamente en uno de los objetivos. Es así como en VEGA, la población total es dividida en  $k$  partes iguales (suponiendo que el problema tiene  $k$  objetivos), y en cada una de ellas opera la selección tomando en cuenta sólo una función objetivo. Una vez realizada la selección, las sub-poblaciones se mezclan para obtener una nueva población de tamaño  $N$ , en la cual se aplicarían el resto de los operadores evolutivos. Todo este proceso se repite en cada generación.

La principal ventaja de VEGA es su facilidad de implementación (sólo se necesita modificar el mecanismo de selección de un algoritmo genético simple) y su eficiencia. Sin embargo, un problema evidente de VEGA es que no favorece a los buenos compromisos, sino que prefiere las soluciones mejores en sólo uno de los objetivos. A este problema se le conoce como especiación (por su análogo en genética) y el peligro potencial está en que se podrían tener individuos que Schaffer llamó "middling"<sup>2</sup>. Este problema fue identificado y atacado por Schaffer [Sch85], utilizando restricciones al cruce, al no permitir recombinación entre individuos de la misma sub-población, así como otras heurísticas adicionales durante la selección en vez del uso de la selección aleatoria de individuos del tradicional AG. También se mostró [RPLH89] que si se utiliza selección proporcional, el esquema de VEGA equivale a una combinación lineal de las funciones objetivo. Así mismo, este método no incorpora explícitamente el concepto de dominancia de *Pareto* (de hecho, se opone a él) y no utiliza ningún mecanismo para mantener diversidad en la población.

Algunos usos de VEGA son los siguientes: contaminación de acuíferos [RER94], colocación de actuadores [Rog00], diseño de circuitos combinatorios a nivel de compuertas [CHB00] y manejo de restricciones [SR97, Coe00].

A partir de 1993, diferentes algoritmos evolutivos basados en *Pareto* fueron sugeridos: a) Fonseca y Fleming con el *Multi-Objective Genetic Algorithm* (MOGA), b) Srinivas y Deb con el *Non-dominated Sorting Genetic Algorithm* (NSGA), y c) Horn y Nafpliotis con el *Niched-Pareto Genetic Algorithm* (NPGA), los cuales profundizaron más en esta técnica de optimización multiobjetivo y cuyas características básicas son: a) evaluación de los miembros de una población con base al concepto de no dominancia de Pareto, y b) la preservación de la diversidad de soluciones. La idea de usar asignación de aptitud basada en el concepto de óptimo de Pareto fue sugerida originalmente por David Goldberg [Gol89] a fin de lidiar con las limitantes de VEGA. Goldberg sugirió también el uso de nichos o alguna técnica similar para mantener diversidad. Esto es necesario debido a que, por ruido estocástico, los algoritmos evolutivos tienden a derivar hacia una solución única independientemente de la distribución inicial de la población.

El MOGA de Fonseca y Fleming [FF93] es un algoritmo que implementa una variante de la jerarquización de *Pareto* propuesta por Goldberg y representa un esquema en el cual la jerarquía de cierto individuo depende del número de individuos en la población actual que lo dominan, como se muestra en la siguiente expresión:

$$\text{jerarquía}(x_i, t) = 1 + p_i^{(t)} \quad (1)$$

donde  $p_i^{(t)}$  es el número de individuos en la generación actual que dominan a  $x_i$

---

<sup>2</sup> Por "middling", Schaffer se refirió a un individuo con eficiencia aceptable, sobre el promedio, pero no excelente para cualquiera de las funciones objetivo.

en la generación  $t$ . Como se ve, todos los individuos no dominados tendrán jerarquía 1, mientras que a los dominados se les penaliza de acuerdo a la densidad de población de la región correspondiente de la superficie de soluciones compromiso.

A continuación de la jerarquización de los individuos de la población, se asignan las aptitudes por medio de una interpolación de las jerarquías, para ello se siguen los siguientes pasos [FF93]:

- a) Ordenar la población con base a las jerarquías de los individuos.
- b) Asignar aptitudes a los individuos interpolando desde el mejor (jerarquía 1) hasta el peor (jerarquía  $n \leq M$ , donde  $M$  es el tamaño de la población) usando el esquema propuesto por Goldberg (Gol89). La interpolación es usualmente lineal, pero puede no serlo.
- c) Promediar las aptitudes de los individuos con la misma jerarquía, de forma que todos ellos sean muestreados de la misma forma. Este procedimiento mantiene constante la aptitud global de la población a la vez que mantiene una presión de selección adecuada.

Desde el punto de vista de Goldberg y Deb [GD91], es probable que este tipo de asignación de la aptitud produzca una presión selectiva grande que podría provocar una convergencia prematura. Para evitar esto, Fonseca y Fleming usaron el método de formación de nichos para distribuir la población sobre la región del óptimo de *Pareto*, pero en este caso sugirieron compartir la aptitud en el espacio de las funciones objetivo y no en el espacio de las variables de decisión. Asimismo, MOGA utiliza restricciones al cruce.

MOGA tiene las mismas desventajas que la jerarquización de Pareto "pura". La principal desventaja la constituye la escalabilidad/complejidad porque el algoritmo de jerarquización es  $O(kM^2)$ , donde  $k$  son las funciones objetivo y  $M$  el tamaño de la población. Su mayor ventaja es la amplia literatura al respecto. También ha mostrado el mejor desempeño de los algoritmos evolutivos multiobjetivo de primera generación. Algunos usos de esta técnica son los siguientes: Co-síntesis de sistemas embebidos de hardware-software [DJ98], diseño de controladores magnéticos [SCFG97], diagnóstico de fallas [Mar97], optimización de armaduras planas [NA99] y manejo de bosques [DdWdB01]. Entre algunos esfuerzos por extender el MOGA están su hibridación con redes neuronales [DRFF00] y la agregación de elitismo [DdWdB01].

Por otro lado, Srinivas y Deb [SD93] propusieron el NSGA que utiliza sin cambios la idea de la jerarquización de Goldberg que ahora se basa en la clasificación de individuos (mediante un criterio de no-dominancia distinto al de MOGA) en varias capas o frentes que ellos llaman ondas (*waves*), es decir, mediante fronteras de no-dominancia. La clasificación consiste en agrupar a todos los individuos no dominados en un frente, con un valor de aptitud igual para todos

los individuos. Para ello se usa el siguiente procedimiento: los individuos que son no dominados con respecto a toda la población son eliminados y se les asigna una "falsa" aptitud, la cual es proporcional al tamaño de la población, para así proporcionar un potencial reproductivo igual para todos los individuos de este frente. Posteriormente, se obtienen los individuos no dominados de la población restante (o sea, de la segunda capa) y se les asigna una aptitud "falsa" menor que la asignada a los individuos de la primera capa. Luego se les elimina de la población. El proceso continúa hasta que todos los individuos de la población hayan sido clasificados. Puesto que los individuos en el primer frente tienen el valor de aptitud mayor, consiguen siempre más copias que el resto. Esto permite buscar las regiones no dominadas, consiguiéndose una convergencia rápida de la población hacia tales regiones.

Para mantener diversidad, el NSGA usa compartición de aptitud (en el espacio de las variables de decisión), la cual se aplica sobre los valores "falsos" de aptitud asignados a cada individuo.

El NSGA tiene también las mismas desventajas que la jerarquización de *Pareto* "pura". En la práctica, el desempeño del NSGA resultó ser el más pobre de entre los algoritmos más representativos de la primera generación (o sea, MOGA y el NPGA). Además, era también el más costoso (computacionalmente hablando). Sin embargo, con este método, al igual que con los otros, puede resolverse cualquier número de objetivos [SD94] y pueden atacar tanto problemas de maximización como de minimización. Algunos usos de esta técnica son los siguientes: control de la calidad del agua [RMG01], diseño de sistemas de control [BW01], diseño de constelaciones [MCH98] y problemas de dinámica de fluidos [MLDP99].

A diferencia de los algoritmos antes descritos que jerarquizan la población completa, Horn y Nafpliotis [HN93] desarrollaron un AGMO, denominado NPGA, cuya idea básica es el uso de la selección mediante torneo, pero se elige a los ganadores del torneo con base a no dominancia de *Pareto* (medida con respecto a una muestra de la población) y no con base a la aptitud. De este modo se evita jerarquizar toda la población.

En el NPGA, cuando se enfrentan dos individuos en un torneo, se comparan ambos contra una fracción de la población (Horn y Nafpliotis utilizan 10%) y, si uno resulta dominado y el otro no, gana el no dominado; pero si los dos son dominados o los dos son no dominados, entonces se realiza un conteo de nichos [GR87], y gana el individuo que esté en una región menos poblada. Es decir, en caso de empates se recurre a un esquema de compartición de aptitud tanto en el espacio de las funciones objetivo como en el de las variables de decisión (se usa *equivalence class sharing*). El tamaño de la población es considerablemente más grande frente a otros enfoques usados para que el ruido del método de selección pueda tolerarse por los nichos surgidos en la población [FF94].

El NPGA es un método muy eficiente, aunque requiere más parámetros que los algoritmos evolutivos multiobjetivo tradicionales. Sin embargo, en los estudios comparativos realizados entre algoritmos de primera generación [ZVL01], normalmente el NPGA resultaba mejor que el NSGA, pero no mejor que el MOGA. Algunas aplicaciones de este método son las siguientes: análisis de espectros experimentales [GLM00], selección de características [EHM00], diseño de sistemas tolerantes a fallas [Sch95b] y diseño de sistemas de tráfico [HG97].

Los AEMOs anteriormente descritos han demostrado ser eficientes en la obtención de múltiples soluciones no dominadas para varios problemas de ingeniería. No obstante, algunos investigadores han sugerido la introducción del elitismo para mejorar sus propiedades de convergencia, por lo tanto, han dado paso a la siguiente generación de AEMOs.

**2. Segunda Generación:** Se introduce el concepto de elitismo en dos formas principales: usando selección ( $\mu + \lambda$ ) y usando una población secundaria. Se busca vencer la complejidad de la jerarquización de *Pareto* ( $O(kM^2)$ ) y de las técnicas tradicionales de nichos ( $O(M^2)$ ). Desde finales de los 1990s son los más usados y entre ellos están *Strength Pareto Evolutionary Algorithm* (SPEA) [ZT98, ZT99], SPEA2 [ZLT01], NSGA-II [DAPM00, DPAM02], NPGA2 [EMH01], *Multiobjective Optimization with Messy GA* (MOMGA) [VL00b], MOMGA-II [ZLV00, ZVL01], PAES [KC99a], *Pareto Envelope-based Selection Algorithm* (PESA) [CKO00], PESA II [CJKO01], micro-GA [CT01a, CT01b], micro-GA2 [TC03], etc.

El SPEA marca el inicio de una segunda generación de AEMOs al introducir el uso de elitismo. Se ha podido demostrar matemáticamente que el elitismo es necesario para garantizar convergencia de un AEMO al verdadero frente de *Pareto*. Este algoritmo fue introducido por Zitzler y Thiele en 1998 [ZT98] como un método que integra diferentes AEMOs, buscando combinar lo mejor de cada uno de ellos. Además del elitismo, el SPEA se diferencia de otros AEMOs de primera generación (MOGA, NSGA, etc) en la manera de la asignación de la aptitud. Este método utiliza un archivo que contiene las soluciones no dominadas encontradas a lo largo del proceso evolutivo (población externa de no dominados). Este conjunto de soluciones no dominadas se suele llamar "conjunto elite" y le permite al algoritmo cubrir de un modo más adecuado el frente de Pareto. Para cada individuo en el sistema externo, se calcula un valor de "fortaleza" (*strength*) que es similar a la jerarquización de MOGA ya que es proporcional al número de soluciones a las que un individuo domina. Luego, la aptitud de cada individuo de la población depende del valor de fortaleza de todos los individuos en el archivo externo que lo dominen [Coe03] (por lo tanto todas las soluciones externas participan en la selección) y se realiza una serie de torneos binarios. Adicionalmente, se usa una técnica de clustering llamada *average linkage method* [Mor80] para mantener diversidad.

El SPEA presenta una desventaja frente a otros AEMOs en términos de



eficiencia, debido al tiempo necesario para la aplicación del algoritmo de clustering en el conjunto elite. Experimentos de comparación entre NSGA y SPEA dan mejores resultados para SPEA [ZDT00, ZT99]. Sin embargo, los resultados de NSGA+Elitismo (NSGA-II) son similares a SPEA [ZDT00]. Algunas aplicaciones de este método son: exploración de compromisos en implementaciones de software para algoritmos DSP [ZTB99], planeación de tratamientos médicos [PM01], asignación en instalaciones radiológicas [LMBZ01] y rehabilitación de sistemas de agua potable [CRFC+03].

Una versión revisada de SPEA, denominada SPEA2, fue propuesta por Zitzler y sus colegas en 2001. El SPEA2 [ZLT01] tiene las siguientes diferencias principales con respecto a su precursor: a) incorpora una estrategia de grano fino para asignar aptitud, la cual toma en cuenta la cantidad de individuos que cada solución domina y la cantidad de individuos que la dominan, b) utiliza la técnica del "vecino más cercano" para la valoración de la densidad, dirigiendo la búsqueda en forma más eficiente; y c) tiene un esquema de truncamiento de archivo que garantiza la preservación de soluciones frontera. Algunos usos de esta técnica son: control del tamaño del código y reducción de *bloat* en programación genética [BBTZ01] y diseño de aeronaves [WB03].

Por su parte, PAES propuesto por Knowles y Corne [KC99a], conceptualmente hablando, es el método más simple posible y usa una estrategia evolutiva  $(1 + 1)$ , es decir, de 2 miembros (sólo padre que es mutado para generar un hijo) adicionada con un archivo externo en el que se van almacenando los individuos no dominados encontrados a lo largo del proceso evolutivo. Cuando se encuentra un individuo no dominado es comparado con los individuos del archivo externo, y en caso de que nuevamente sea no dominado, puede entrar al mismo archivo. El mecanismo de PAES para mantener la diversidad consiste en una rejilla adaptativa, que es computacionalmente más eficiente que los métodos de nichos. El uso de la rejilla adaptativa permite retener a los individuos no dominados obtenidos y distribuirlos uniformemente.

Knowles y Corne también experimentaron con estrategias evolutivas  $(1+\lambda)$  y  $(\mu+\lambda)$ , pero aseguran que no hay mejoras significativas y sin embargo sí un aumento en el esfuerzo computacional necesario [KC00]. Entre las aplicaciones de PAES están las siguientes: problemas de telecomunicaciones [KC99b, KOC00] y manejo de bases de datos distribuidas adaptativas [KCO00].

Por otro lado, Corne y otros [CKO00] propusieron un algoritmo, llamado PESA, en el que se tiene una población pequeña principal (interna), y una población secundaria más grande, que corresponde con el archivo externo mencionado antes (como PAES). PESA usa la rejilla adaptativa de PAES para mantener diversidad. Sin embargo, en este caso, el mecanismo de selección está basado en la medida de agrupamiento (*crowding*) usada por la rejilla auto-adaptativa. Esta misma medida se usa para decidir que soluciones ingresan al archivo externo.

El funcionamiento de PESA se describe a continuación. Durante cada iteración del algoritmo, se seleccionan aleatoriamente individuos del archivo externo, y de ellos se producen los nuevos individuos que integrarán la población principal; cuando ésta población principal se llene, los individuos no dominados se incorporarán al archivo secundario. Este algoritmo ha sido aplicado a problemas de telecomunicaciones [CKO00].

Posteriormente apareció PESA-II, de Corne y otros [CJKO01], cuya diferencia principal con PESA es la selección basada en regiones. En este esquema de selección, la unidad de selección es la hiperrejilla en vez del individuo, lo que significa que la selección considera regiones y no individuos. Esto busca reducir el costo computacional. En una fase siguiente se elige al azar un individuo de cada región seleccionada. Este algoritmo también ha sido usado en problemas de telecomunicaciones [CJKO01].

Deb y sus colegas [DAPM00] propusieron una versión revisada del NSGA, llamada NSGA-II, que además de contar con el uso de elitismo es mucho más eficiente computacionalmente hablando que su predecesor y constituye un algoritmo altamente competitivo en convergencia al *Pareto*, aunque parece que sigue teniendo problemas para generar regiones aisladas del frente de *Pareto* (problemas exploratorios) sobre todo conforme se incrementa el número de funciones objetivo [ZVL01], así como para funcionar adecuadamente con representación binaria [CVL02]. El NSGA-II [DAPM00] utiliza un operador de *crowding* (medida de concentración) para mantener la diversidad que no requiere parámetros, en vez de usar nichos. Asimismo, no utiliza memoria externa como los algoritmos anteriores (SPEA y SPEA2). El mecanismo elitista consiste en elegir los mejores  $P$  individuos de la unión de las poblaciones padre e hijo. Para ello, el NSGA-II utiliza un esquema de selección más (+) en el cual la población de padres se compara con la población de hijos.

El funcionamiento del NSGA-II [DPAM02] se describe a continuación. En cada generación, se crea un conjunto mediante la unión de la actual población y la creada mediante selección, cruce y mutación. De este conjunto se extraen los diferentes frentes (agrupados según el número de soluciones que los dominan). El frente  $F_1$  coincide con el actual frente de *Pareto*. La nueva población se crea incluyendo los frentes (de mejor a peor) hasta alcanzar el tamaño máximo. Si es necesario, el último frente se trunca atendiendo al orden basado en el *crowding*. La medida de *crowding* se utiliza para seleccionar las soluciones más dispersas entre los individuos del último frente utilizado en la nueva población. Cuanto mayor sea la distancia de *crowding* de una solución al resto de su frente mejor, ya que hay menos concentración en esa zona.

Algunos usos del NSGA-II son los siguientes: optimización de formas [DG01], diseño de sistemas de seguridad [GGW03], extrusión de polímeros [GPVC02], manejo de la calidad del agua [DR03] y terapias de radiación

[LSMB03].

Van Veldhuizen y Lamont [VL00b] propusieron la versión multiobjetivo del algoritmo desordenado *messy GA* [Deb91], denominada MOMGA. En la primera fase se generan de manera exhaustiva todos los bloques constructores hasta cierto tamaño pre-especificado usando un procedimiento determinista conocido como inicialización parcialmente enumerativa. La siguiente fase realiza la selección por medio de torneo en la población y, de ser necesario, se reduce su tamaño. Por último, la fase yuxtaposicional genera la población nueva por medio de un operador de recombinación por corte y empalme. Algunas aplicaciones de este método son las siguientes: diseño de controladores [HBP02] y funciones de prueba tradicionales [VL00b].

MOMGA-II fue propuesto por Zydallis y otros [ZVL01] y es la versión multiobjetivo del algoritmo genético desordenado rápido [GDKH93]. En este caso, los autores extendieron el *fast-messy GA*, el cual consta de tres fases: 1) Fase de inicialización, 2) Filtrado de bloques constructores, y 3) Fase yuxtaposicional. La principal diferencia del MOMGA-II con respecto a su antecesor radica en las primeras dos fases. La fase de inicialización utiliza un proceso de inicialización probabilístico que crea un número controlado de clones de los bloques constructores de un cierto tamaño predeterminado. La segunda fase, llamada de filtrado de bloques constructores, consiste en reducir la población mediante un filtrado, de forma que se almacenen los mejores bloques constructores; además, en esta fase se realiza el torneo de selección. La tercera fase es la misma que la de MOMGA. Este método ha sido aplicado a funciones de prueba tradicionales [ZVL01] y ha dado buenos resultados en la mayoría de los casos en comparación con otros AEMOs tales como NSGA, MOGA, NPGA, PAES, SPEA, NSGA-II y MOMGA, pudiendo ser considerado como un algoritmo eficiente y efectivo para atacar aplicaciones reales [ZVL01].

Posteriormente Erickson y otros [EMH01], realizaron en el 2001 una segunda versión del algoritmo NPGA. El NPGA2 jerarquiza toda la población antes de realizar los torneos. Para ello, usa la jerarquización de *Pareto* pero mantiene la selección mediante torneo del NPGA original. La segunda diferencia importante es que se usa una nueva forma de compartición de aptitud en la que se calculan los conteos de nichos utilizando individuos de la siguiente generación, aunque se encuentre incompleta, en vez de usar los de la generación actual. A esta forma de realizar conteo de nichos se le llama *continuously updated fitness sharing* y fue propuesta previamente por Oei y otros [OGC91]. El NPGA2 ha sido aplicado en la contaminación de acuíferos [EMH01].

Por su parte, Coello y Toscano [CT01a, CT01b] desarrollaron un micro-GA, que es un algoritmo que también tiene una población principal pequeña. Un micro algoritmo genético es un algoritmo genético con un tamaño de población  $\leq 5$  individuos. La clave del micro algoritmo genético es realizar un proceso de reinicialización una vez que se alcanza convergencia nominal.

Este algoritmo cuenta con un archivo llamado memoria de la población, y está dividido en una parte reemplazable y una parte no reemplazable, de la cual se eligen aleatoriamente los individuos que formarán la población principal. Esta población pequeña es la que utiliza el algoritmo genético con operadores normales, que cuando converge proporciona los individuos no dominados que entrarán en la memoria externa (archivo externo). Algunos de los individuos en esta memoria externa entrarán periódicamente en la parte reemplazable de la memoria de la población. En total, el micro-GA utiliza tres tipos de elitismo y la rejilla adaptativa de PAES.

El micro-GA [CT01a, CT01b] es uno de los AEMOs más rápidos de la actualidad, superando en varios órdenes de magnitud a algoritmos muy eficientes tales como el NSGA-II. No obstante, el mayor inconveniente que presenta es que requiere un elevado número de parámetros (ocho, de los cuales al menos 3 son críticos para su desempeño). Algunas aplicaciones de esta técnica son las siguientes: diseño de aviones [CCA03], diseño estructural [CVL02] y particiones de sistemas de hardware/software [FMSZ03].

Buscando eliminar la mayor desventaja del micro-GA original, Toscano y Coello propusieron recientemente el micro-GA2 [TC03], el cual usa mecanismos de adaptación en línea que hacen que el algoritmo no requiera ningún parámetro. El micro-GA2 es capaz incluso de decidir cuándo debe detenerse (no requiere un número máximo de generaciones). El único parámetro que el usuario debe proporcionar es el tamaño de la población secundaria (aunque puede usarse un valor por omisión). Hasta ahora las únicas aplicaciones del micro-GA2 han sido funciones de prueba estándar [Tos05].

Para finalizar, se puede indicar que hoy en día los algoritmos no basados en dominancia de *Pareto* siguen siendo usados en algunos dominios (por ejemplo, optimización combinatoria) con relativo éxito. Asimismo, aún no se tiene una visión clara en el área de los AEs respecto a qué tipo de algoritmos multiobjetivo son más adecuados para qué tipo de problemas.

Además, es necesario señalar que aunque se han abordado un volumen considerable de aplicaciones hasta la fecha [Coe03, FDA04], la optimización evolutiva multiobjetivo (EMOO) todavía tiene mucho que ofrecer en términos de dominios que aún no han sido abordados tales como visión por computador, ajuste de modelos basados en elementos finitos, reconocimiento de patrones, etc. Sin embargo, la nueva fase en la que se encuentra el área amerita mayor dedicación y la generación de ideas más profundas. Posiblemente todavía queden problemas simples por resolver, pero hay que buscarlos con mayor ahínco. Adicionalmente, muchos problemas de dificultad "intermedia" todavía no han sido resueltos adecuadamente. Claramente, el diseño de algoritmos para problemas de alto grado de dificultad será una línea que permanecerá abierta durante algún tiempo. Por lo

tanto, la EMOO en la actualidad presenta muchas oportunidades de investigación y su futuro luce aún prometedor.

## **2. Generación de particiones difusas adecuadas usando algoritmos genéticos multiobjetivo (AGMO3)**

En la experimentación realizada sobre varias aplicaciones [AN05, ANF04, ANF05a, ANF05b, ANF06a, ANF06b] se observó que cuando se utilizaba la función objetivo  $MSE_{entr}$  se obtenían, frecuentemente, granularidades elevadas, que alcanzaban en algunas variables su valor máximo de 9 etiquetas. Este no es el caso en la función objetivo 1-Q, con la que se obtienen granularidades de 6 etiquetas como máximo en las variables. Por otro lado, se confirma que la función objetivo  $MSE_{entr}$  permite, generalmente, obtener modelos FIR con mayor poder de predicción (en comparación con los encontrados usando 1-Q), pero con un costo computacional mayor. Así mismo, se hace patente que la medida de calidad  $Q$  usada por la metodología FIR para determinar el poder de predicción de una máscara no está haciendo un buen trabajo, siendo fundamental investigar medidas alternativas de calidad en un futuro próximo. Es por todo ello, que el propósito principal del método multiobjetivo propuesto en esta investigación es la obtención de modelos FIR con un nivel de complejidad reducido (mínimo número de etiquetas por variable) y una alta precisión en las predicciones (menor error MSE de predicción de una parte del conjunto de datos de entrenamiento,  $MSE_{entr}$ ). Desafortunadamente, es difícil conseguir las dos cosas al mismo tiempo y se requerirá alcanzar un compromiso entre ambos objetivos.

Normalmente, los modelos FIR con muchas etiquetas por variable son modelos complejos, éstos poseen una mayor expresividad pero tienen un menor poder de predicción (poca predictividad). La disminución del poder de predicción cuando aumenta el número de etiquetas de las variables del sistema es debido al problema de la maldición de la dimensionalidad. Si discretizamos los datos del sistema en un número elevado de clases por variable, tendremos un modelo muy expresivo, con un alto nivel de interpretabilidad. Sin embargo, su poder de predicción o predictividad será, probablemente, reducido puesto que el número de estados posibles habrá aumentado considerablemente mientras que el conjunto de datos de entrenamiento se mantiene constante. En esta situación sucederá que el siguiente estado a predecir no se encontrará en la base de reglas patrón, puesto que no se habrá observado previamente. Esto llevará a la situación en que el modelo no pueda predecir ningún estado futuro del sistema y tenga una capacidad nula de predicción. Es así como el proceso de aprendizaje de las particiones difusas para el diseño de un modelo FIR utilizará ahora un AG multiobjetivo, el cual persigue obtener modelos que sean lo más simples posible pero manteniendo el máximo poder de predicción. Para conseguir este balance/equilibrio se consideran los siguientes objetivos:

- *Obtener modelos FIR lo más simples posible:* Para alcanzar este objetivo se penalizan los modelos FIR con niveles de granularidad altos.
- *Obtener modelos FIR lo más precisos posible:* Para lograrlo se minimizará el error de predicción de una parte del conjunto de datos de entrenamiento.

Las secciones siguientes presentan el método propuesto describiendo los componentes principales del AGMO.

### 3. Aspectos principales del método propuesto

#### 3.1 Codificación de las soluciones

Del mismo modo que en [ANF05b], cada cromosoma está compuesto de dos partes que codifican el número de clases de cada variable (granularidad) y la función de pertenencia más adecuada para cada clase:

- Número de clases ( $C_1$ ): La representación genética utilizada es la misma que ha sido descrita en [ANF05b].
- Funciones de pertenencia ( $C_2$ ): Esta representación genética toma en cuenta el número de elementos observados para cada variable y ya ha sido descrita en [ANF05b].

Usaremos, por lo tanto, la misma representación de los cromosomas que se describió en [ANF05b] para el **AG3** y que recordamos aquí:

$$C = C_1 C_2 \quad (2)$$

Si denotamos por  $E_j$  el número de clases de la variable  $j$ , la representación del número de clases para un sistema de  $N$  variables ( $C_1$ ), es definida por:

$$C_1 = (E_1, E_2, \dots, E_N) \quad (3)$$

También, si denotamos por  $D_{ij}$  la proporción de datos para la clase  $i$  de la variable  $j$ , y por  $C_{2j}$  la información de la proporción de datos para todas las clases de la variable  $j$ ,

$$C_{2j} = (D_{1j}, \dots, D_{E_{jj}}) \quad (4)$$

la representación de las funciones de pertenencia ( $C_2$ ) para un sistema de  $N$  variables (incluyendo entradas y salidas) se define como:

$$C_2 = (C_{21}, C_{22}, \dots, C_{2N}) \quad (5)$$

Lógicamente, cada vez que cambie el número de clases y/o la distribución

de los landmarks de una variable por la acción de los operadores genéticos, se tendrá que construir la nueva partición difusa correspondiente.

### **3.2 Población inicial**

La población inicial está compuesta por cuatro bloques. A excepción del primero el resto posee el mismo número de cromosomas. En ninguno de los casos se admiten cromosomas repetidos. El proceso de generación de la misma se describe a continuación:

- En el primer bloque cada cromosoma tiene el mismo número de etiquetas para todas las variables del problema y se consideran funciones de pertenencia distribuidas uniformemente (Método EFP) a lo largo del universo de discurso de la variable.
- En el segundo bloque, dentro de cada cromosoma puede haber distintos valores (escogidos aleatoriamente) para el número de etiquetas de las variables. Las funciones de pertenencia se distribuyen uniformemente (Método EFP) al igual que en el primer bloque considerando la granularidad correspondiente.
- En el tercer bloque se empieza igual que en el primero, cada cromosoma tiene el mismo número de etiquetas para todas las variables del problema. Las funciones de pertenencia no se distribuyen uniformemente, se generan aleatoriamente.
- En el último bloque se comienza igual que en el segundo, cada variable de cada cromosoma puede tener cualquier número de etiquetas. Las funciones de pertenencia se generan aleatoriamente igual que en el tercer bloque.

La generación de esta forma de la población inicial persigue lograr una diversidad apropiada. Aunque los AGs han probado ser robustos y capaces de obtener buenas soluciones en poblaciones que han sido generadas aleatoriamente (grupo cuatro), se puede obtener una mayor convergencia con la distribución propuesta.

### **3.3 Evaluación de los cromosomas**

Para evaluar un cromosoma se llevan a cabo los siguientes pasos:

- 1) Se decodifica la información del cromosoma, construyendo su correspondiente partición difusa en la estructura de datos que utiliza FIR para representar dichas particiones.
- 2) Se ejecuta el proceso de identificación de modelado cualitativo de la

metodología FIR con los datos de entrenamiento, usando la partición construida en el paso anterior. Por lo tanto, se obtiene la máscara asociada a esa partición difusa con la medida más alta de calidad.

3) Se calculan los valores de la función de evaluación multiobjetivo:

- $\Sigma_{eti}$ : Corresponde a la sumatoria de la granularidad de las variables de entrada y de la variable de salida cuya relación causal o temporal se incluye en la máscara óptima. Sin embargo, cabe aclarar aquí que si una variable aparece en la máscara en diferentes instantes de tiempo, su granularidad se contabiliza una única vez.
- $MSE_{entr}$ : Indica el error de predicción de una parte del conjunto de datos de entrenamiento. El error cuadrático medio normalizado en porcentaje (MSE), dado en la ecuación (6), se usa para este fin.

$$MSE = \frac{E \left[ (y(t) - \hat{y}(t))^2 \right]}{y_{var}} \cdot 100\% \quad (6)$$

$\hat{y}(t)$  es la salida predicha,  $y(t)$  la salida del sistema y  $y_{var}$  denota la varianza de  $y(t)$ . La idea es usar parte del conjunto de datos de entrenamiento para identificar el modelo y el resto del conjunto de datos para evaluar la eficiencia (*performance*) de la predicción del modelo encontrado con la partición difusa evaluada. Cabe recordar que el modelo FIR está compuesto por la máscara óptima y la base de reglas patrón (*I/O matrix*). Esto implica que ambos han de ser generados en el proceso de evaluación de una determinada partición difusa cuando se usa esta parte de la función multiobjetivo. El error de predicción de la porción del conjunto de datos de entrenamiento no usado en el proceso de identificación del modelo constituye uno de los componentes de la función de evaluación multiobjetivo. Por lo tanto, es necesario llamar a la función de predicción de la metodología FIR para obtener el costo del cromosoma evaluado. El tamaño de la porción del conjunto de datos de entrenamiento usado para este componente de la función de evaluación multiobjetivo se define con respecto al tamaño del conjunto de datos de entrenamiento disponible. Concretamente se ha especificado que para conjuntos de datos de entrenamiento con un número de instancias superior a 2000 el 25% final se usa para la evaluación de cada cromosoma, si se disponen de entre 400 y 2000 instancias se usa el 20% final de éstas y si es menor o igual a 400 instancias se utilizaría sólo el 8% final.

### 3.4 Operadores genéticos

Para el diseño de los operadores genéticos, se ha considerado la estructura especial de los cromosomas (con dos niveles de información distintos, pero fuertemente relacionados) para obtener una definición adecuada que haga el mejor uso posible de la representación adoptada. A continuación se describen los mismos.



### 3.4.1 Selección

Se ha utilizado el mecanismo de selección multiobjetivo MOGA [FF93], que se basa en la definición *Pareto*-optimalidad. Se dice que una solución *domina* a otra cuando tiene valores mejores o iguales que la otra en todos los objetivos salvo en uno, donde tiene un valor estrictamente mejor. El conjunto óptimo de *Pareto* está compuesto por todas las soluciones no dominadas del problema. El operador consta de cinco pasos:

1. A cada individuo se le asigna un rango que es igual al número de soluciones de la población que lo dominan más uno (las soluciones no dominadas tienen rango 1).
2. La población se ordena de forma ascendente de acuerdo con su rango.
3. Se le asigna una probabilidad de selección a cada individuo que depende de su rango (menor rango, mayor probabilidad).
4. Se calcula la media de la probabilidad de selección de cada clase de equivalencia (grupo de cromosomas con el mismo rango) para que todos los individuos de igual rango tengan la misma probabilidad de reproducirse en la siguiente generación.
5. Se crea la nueva población mediante el muestreo universal estocástico de Baker [Bak87].

### 3.4.2 Cruce

Se utilizan los mismos operadores de cruce descritos en [ANF05b], éstos toman en cuenta las características de los dos cromosomas padres implicados en el cruce:

- *Cruce cuando los dos padres tienen la misma granularidad en cada una de las variables:* Si los dos cromosomas tienen los mismos valores en  $C_1$  (cada variable tienen el mismo número de etiquetas en ambos padres), se puede suponer que la búsqueda genética ha localizado una zona prometedora del espacio que conviene explotar. Para ello, se utiliza el operador de cruce aritmético no uniforme [Mic96] sobre  $C_2$ , manteniendo el valor de  $C_1$  en ambos descendientes. El funcionamiento de este operador se estableció en [ANF05b].
- *Cruce cuando los padres tienen distinto número de etiquetas en alguna de las variables:* En este segundo caso, parece interesante utilizar la información contenida en ambos padres para explorar el espacio de búsqueda y tratar de localizar nuevas zonas prometedoras. El funcionamiento del operador de cruce es sencillo, se selecciona aleatoriamente un punto de corte dentro de  $C_1$ , y se cruzan las dos partes  $C_1$  y  $C_2$  de ambos cromosomas de acuerdo al operador clásico de cruce [Mic96].

A continuación, se recuerda como funciona este cruce mediante una representación gráfica. Sean

$$\begin{aligned} C_t &= (E_1, \dots, E_p, E_{p+1}, \dots, E_N, C_{21}, \dots, C_{2p}, C_{2p+1}, \dots, C_{2N}) \\ C'_t &= (E'_1, \dots, E'_p, E'_{p+1}, \dots, E'_N, C'_{21}, \dots, C'_{2p}, C'_{2p+1}, \dots, C'_{2N}) \end{aligned} \quad (7)$$

dos individuos que se van a cruzar en el punto p, los dos descendientes resultantes son:

$$\begin{aligned} C_t &= (E_1, \dots, E_p, E'_{p+1}, \dots, E'_N, C_{21}, \dots, C_{2p}, C'_{2p+1}, \dots, C'_{2N}) \\ C'_t &= (E'_1, \dots, E'_p, E_{p+1}, \dots, E_N, C'_{21}, \dots, C'_{2p}, C_{2p+1}, \dots, C_{2N}) \end{aligned} \quad (8)$$

De esta forma, el proceso completo de recombinación permitirá al AGMO seguir una tasa adecuada de exploración y explotación del espacio de búsqueda.

#### 8.3.4.3 Mutación

Los operadores de mutación seleccionados son los mismos que se han descrito en [ANF05b]:

- *Mutación en  $C_1$* : El operador de mutación seleccionado es el propuesto por [Thr91]. Si el gen a mutar es de  $C_1$ , se aumenta o disminuye en una unidad la granularidad asociada (la decisión es tomada de manera aleatoria). Cuando el valor que se pretende cambiar es el mínimo (2) ó el máximo (9), se realiza el único cambio posible, esto es, incrementar o decrementar en uno la granularidad, respectivamente. Por supuesto, un cambio en  $C_1$  origina una inconsistencia en  $C_2$ , donde se encuentran almacenados los conjuntos difusos asociados al número de clases (funciones de pertenencia) que tenía antes esa variable. Por tanto, cuando se cambia la granularidad de una variable en  $C_1$  se cambia su parte correspondiente en  $C_2$ , construyendo una partición difusa uniforme (Método EFP) con el nuevo número de clases del cromosoma.
- *Mutación en  $C_2$* : Debido a la naturaleza de los valores almacenados en esta parte del cromosoma, el operador de mutación seleccionado es similar a uno propuesto por Thrift en [Thr91] que ya ha sido descrito en [ANF05b].

## 4. Comentarios

Aunque el método propuesto en esta investigación se basa en un mecanismo de selección MOGA que está enmarcado en la primera generación de AGs multiobjetivo, y por lo tanto, no incorpora una selección elitista, creemos que éste puede permitir obtener unos buenos resultados en un primer intento y se prevé que

más adelante estos resultados podrían mejorarse utilizando un AG multiobjetivo más eficaz.

La ventaja principal del **AGMO3** estriba en que en una sola ejecución (con un tiempo similar al necesitado por los AGs implementados anteriormente), permite obtener un conjunto de soluciones (unas más simples y otras más precisas para poder escoger) al alcanzar el criterio de parada que se haya estipulado. De este modo, no es necesario ejecutarlo tantas veces como los métodos **AG1**, **AG2** y **AG3** contemplados previamente en trabajos previos [ANF04, ANF06a, ANF05b].

## 5. Caso de estudio: Concentraciones de ozono en la Ciudad de México

En este apartado se presentan los resultados obtenidos por el **AGMO3** para el problema de predicción de concentraciones de ozono en la Ciudad de México descrito en [ANF06b]. Cabe destacar que el AG multiobjetivo propuesto en esta investigación se encuentra aún en fase de ajuste y pruebas.

Se han realizado cuatro ejecuciones (con distinto valor de la semilla) para cada criterio de parada. Los resultados obtenidos para cada una de las ejecuciones y criterio de parada se muestran en el Apéndice. Se estudia la función multiobjetivo descrita en la sección 3.3. En esta aplicación medioambiental el último 25% de la señal de entrenamiento se usa para la evaluación de la función multiobjetivo y el 75% inicial de la señal se usa para la obtención de los modelos FIR (máscaras y bases de reglas patrón).

Del mismo modo que en el estudio previo [ANF06b], se ha trabajado bajo la perspectiva del modelado del comportamiento del ozono durante un mes específico del año [NM06]. Sin embargo, en este caso solo se ha seleccionado el mes de Enero que pertenece a la época seca. Asimismo, se utiliza la máscara candidata (ecuación 3) que se presenta a continuación:

$t \backslash x$	HD	HR	TMP	DV	VV	03
$t - 2\delta t$	-1	-1	-1	-1	-1	-1
$t - \delta t$	-1	-1	-1	-1	-1	-1
$t$	-1	-1	-1	-1	-1	+1

(3)

Los valores establecidos para el AG multiobjetivo se presentan en la *Tabla 1*. Las probabilidades de cruce y de mutación son establecidas de acuerdo a [Gre86]. Se emplearon computadoras Pentium IV (2.66 GHz) para realizar todas las simulaciones. Hay que señalar que se probó con una población inicial de 50

individuos pero no se obtenía mucha diversidad en el conjunto de soluciones. Los mejores resultados obtenidos por el AG multiobjetivo se presentan en las *Tablas 2 a 6*.

**Tabla 1.** Parámetros del AG multiobjetivo para el problema de la estimación de concentraciones locales máximas de ozono en México DF.

Parámetro	Valor
<i>Tamaño de la población</i> (# individuos)	100
<i>Probabilidad de cruce</i>	0.6
<i>Probabilidad de mutación</i>	0.1
<i>Criterio de parada</i> (# evaluaciones de cromosomas)	{2000, 4000, 8000, 16000}

### 5.1 Modelos obtenidos por el AGMO3

En esta sección se presentan los resultados obtenidos por el **AGMO3** propuesto en esta investigación sobre el problema de modelado de concentraciones de ozono en la ciudad de México. Las *Tablas 2 a 5* muestran los resultados obtenidos por criterio de parada (número de evaluaciones de cromosomas hechas por el AG multiobjetivo). Las tablas están organizadas como se describe a continuación. La primera columna se divide en dos secciones. La sección *A* corresponde al mejor resultado obtenido por el **AGMO3** (valor mínimo de cada parte que conforma la función multiobjetivo, es decir, el menor error  $MSE_{entr}$  y la menor sumatoria de etiquetas conseguidas) en cada una de las 4 ejecuciones. La sección *B* corresponde a los mejores resultados de predicción del **AGMO3** mostrados para las 4 ejecuciones. La segunda columna indica el número de ejecución del **AGMO3**. La tercera y cuarta columnas presentan la solución sugerida por el **AGMO3** (número de clases y distribución de landmarks) para las variables de entrada (HD, HR, TMP, DV, VV) y la variable de salida ( $O_3$ ) cuando se cumple el criterio de parada estipulado. El número de elementos de la proporción de datos corresponde al número de clases para esa variable. La quinta columna contiene por un lado, la máscara óptima (en notación de posición) encontrada por FIR para esa partición difusa. Por el otro lado, contiene el valor de la sumatoria de la granularidad de las variables de entrada y de salida que se incluye en esa máscara, pero sin tener en cuenta más de una relación temporal. Es decir, si una variable aparece en la máscara en diferentes instantes de tiempo, se contabiliza una única vez. La sexta columna muestra la calidad asociada a la máscara óptima encontrada. La séptima columna corresponde al valor del error  $MSE_{entr}$  de la función multiobjetivo. La última columna muestra el error de predicción RMS para el conjunto de datos de prueba ( $RMS_{prue}$ ).

En la *Tabla 6* se presenta el tiempo consumido, número de generaciones y número de soluciones no dominadas obtenidas por ejecución y criterio de parada del AG multiobjetivo para el mes considerado.

**Tabla 2.** Resultados de la granularidad y las funciones de pertenencia para la estimación de la concentración de ozono en México usando el AG multiobjetivo. Criterio de parada = 2000 evaluaciones. Mes Enero.

# ejec.	Gran.	Proporción de datos	Opt. Mask / $\Sigma$ etiquetas	Q	MSE <sub>entr</sub>	RMS <sub>prue</sub>
A	1 (2,2,2,3,2,2)	HD:(0.73,0.27) ; HR:(0.70,0.30) ; TMP:(0.29,0.71) DV:(0.33,0.33,0.34) ; VV:(0.26,0.74) O3:(0.71,0.29)	(4,7,12,16,18) 7 etiq.	0.7706	40.7664	0.0196
	(4,2,8,4,7,2)	HD:(0.41,0.10,0.10,0.39) ; HR:(0.29,0.71) TMP:(0.12,0.12,0.12,0.12,0.13,0.13,0.13,0.13) DV:(0.3,0.17,0.05,0.48) VV:(0.27,0.28,0.05,0.11,0.13,0.07,0.09) O3:(0.37,0.63)	(12,18) 2 etiq.	0.6323	141.2987	0.0298
	2 (5,6,2,2,8,3)	HD:(0.15,0.15,0.15,0.40,0.15) HR:(0.16,0.16,0.17,0.17,0.17,0.17) TMP:(0.27,0.73) ; DV:(0.33,0.67) VV:(0.12,0.12,0.12,0.12,0.13,0.13,0.13,0.13) O3:(0.33,0.33,0.34)	(1,12,18) 8 etiq.	0.6505	43.0374	0.0215
	(3,3,3,2,3,2)	HD:(0.05,0.84,0.11) ; HR:(0.52,0.11,0.37) TMP:(0.07,0.86,0.07) ; DV:(0.50,0.50) VV:(0.29,0.08,0.63) ; O3:(0.64,0.36)	(12,18) 2 etiq.	0.6731	127.629	0.0407
	3 (6,3,2,5,3,3)	HD:(0.16,0.16,0.17,0.17,0.17,0.17) HR:(0.33,0.33,0.34) ; TMP:(0.73,0.27) DV:(0.20,0.20,0.20,0.20,0.20,0.20) VV:(0.33,0.33,0.34) ; O3:(0.33,0.33,0.34)	(1,12,18) 9 etiq.	0.6492	43.1024	0.0216
	(2,3,3,3,3,2)	HD:(0.16,0.84) ; HR:(0.33,0.33,0.34) TMP:(0.33,0.33,0.34) ; DV:(0.33,0.33,0.34) VV:(0.33,0.33,0.34) ; O3:(0.66,0.34)	(12,18) 2 etiq.	0.675	125.9375	0.0404
	4 (7,4,4,3,4,9)	HD:(0.05,0.05,0.07,0.13,0.35,0.27,0.08) HR:(0.25,0.25,0.25,0.25) TMP:(0.07,0.44,0.44,0.05) ; DV:(0.36,0.41,0.23) VV:(0.19,0.29,0.32,0.20) O3:(0.09,0.10,0.13,0.10,0.09,0.10,0.09,0.22,0.08)	(1,12,18) 16 etiq.	0.4822	43.1197	0.0221
	(2,2,2,2,3,2)	HD:(0.32,0.68) ; HR:(0.51,0.49) TMP:(0.21,0.79) ; DV:(0.47,0.53) VV:(0.33,0.33,0.34) ; O3:(0.75,0.25)	(12,18) 2 etiq.	0.7131	116.3262	0.0386
	1 (2,2,2,4,2,2)	HD:(0.73,0.27) ; HR:(0.70,0.30) TMP:(0.29,0.71) ; DV:(0.25,0.25,0.25,0.25) VV:(0.26,0.74) ; O3:(0.71,0.29)	(7,12,16,18) 8 etiq.	0.7721	61.9794	0.0175
	2 (2,2,2,3,2,2)	HD:(0.40,0.60) ; HR:(0.24,0.76) TMP:(0.27,0.73) ; DV:(0.33,0.33,0.34) VV:(0.30,0.70) ; O3:(0.58,0.42)	(1,12,13,18) 4 etiq.	0.6907	59.2999	0.0175
B	3 (3,2,6,5,2,6)	HD:(0.33,0.33,0.34) ; HR:(0.22,0.78) TMP:(0.09,0.51,0.14,0.05,0.12,0.09) DV:(0.11,0.20,0.41,0.13,0.15) ; VV:(0.81,0.19) O3:(0.29,0.12,0.08,0.10,0.14,0.27)	(1,12,18) 9 etiq.	0.5985	50.504	0.0194
	4 (9,5,2,4,3,5)	HD:(0.05,0.21,0.07,0.06,0.05,0.14,0.18,0.18,0.06) HR:(0.11,0.46,0.05,0.26,0.12) TMP:(0.48,0.52) ; DV:(0.3,0.32,0.26,0.12) VV:(0.37,0.50,0.13) ; O3:(0.09,0.36,0.05,0.17,0.33)	(3,12,15,18) 7 etiq.	0.6028	60.6752	0.0204

**Tabla 3.** Resultados de la granularidad y las funciones de pertenencia para la estimación de la concentración de ozono en México usando el AG multiobjetivo. Criterio de parada = 4000 evaluaciones. Mes Enero.

# ejec.	Gran.	Proporción de datos	Opt. Mask / Σ etiquetas	Q	MSE <sub>entr</sub>	RMS <sub>prue</sub>		
A	1	(3,2,2,2,4,2)	HD:(0.20,0.61,0.19) ; HR:(0.52,0.48) TMP:(0.76,0.24) ; DV:(0.58,0.42) ; VV:(0.25,0.25,0.25,0.25) ; O3:( 0.55,0.45)	(12,13,14,17,18) 11 etiq.	0.6517	39.8551	0.0199	
		(3,2,3,2,2,2)	HD:(0.33,0.33,0.34) ; HR:(0.67,0.33) TMP:(0.33,0.33,0.34) ; DV:(0.38,0.62) VV:(0.35,0.65) ; O3:(0.81,0.19)	(6,12,18) 2 etiq.	0.7478	106.5236	0.0393	
	2	(6,2,5,8,3,3)	HD:(0.16,0.16,0.17,0.17,0.17,0.17) ; HR:(0.50,0.50) TMP:(0.20,0.20,0.20,0.20,0.20) DV:(0.12,0.12,0.12,0.12,0.13,0.13,0.13,0.13) VV:(0.33,0.33,0.34) ; O3:(0.33,0.33,0.34)	(1,12,18) 9 etiq.	0.6492	43.1024	0.0216	
		(3,2,2,2,2,2)	HD:(0.33,0.33,0.34) ; HR:(0.57,0.43) TMP:(0.45,0.55) ; DV:(0.39,0.61) VV:(0.32,0.68) ; O3:(0.82,0.18)	(6,12,18) 2 etiq.	0.7545	119.4284	0.0387	
	3	(2,2,4,2,4,2)	HD:(0.76,0.24) ; HR:(0.52,0.48) TMP:(0.25,0.25,0.25,0.25) ; DV:(0.21,0.79) VV:(0.25,0.25,0.25,0.25) ; O3:(0.61,0.39)	(12,13,14,17,18) 10 etiq.	0.7208	42.8004	0.0211	
		(2,8,3,8,4,2)	HD:(0.24,0.76) HR:(0.06,0.05,0.05,0.17,0.28,0.05,0.22,0.12) TMP:(0.07,0.59,0.34) DV:(0.14,0.27,0.16,0.05,0.05,0.05,0.21,0.07) VV:(0.46,0.12,0.21,0.21) ; O3:(0.77,0.23)	(12,18) 2 etiq.	0.7185	114.1194	0.0381	
	4	(5,3,7,8,4,2)	HD:(0.05,0.26,0.05,0.19,0.45) ; HR:(0.28,0.53,0.19) TMP:( 0.26,0.10,0.22,0.19,0.07,0.09,0.07) ; DV:(0.07,0.12,0.16,0.21,0.05,0.18,0.12,0.09) VV:(0.15,0.19,0.33,0.33) ; O3:(0.60,0.40)	(1,12,18) 7 etiq.	0.7397	43.1676	0.0221	
		(5,3,7,7,4,2)	HD:(0.10,0.05,0.05,0.27,0.53) ; HR:(0.45,0.20,0.35) TMP:(0.23,0.06,0.18,0.14,0.05,0.29,0.05) DV:(0.14,0.14,0.14,0.14,0.14,0.15,0.15) VV:(0.15,0.19,0.33,0.33) ; O3:(0.79,0.21)	(12,18) 2 etiq.	0.7261	111.6386	0.0375	
	B	1	(3,2,2,2,2,2)	HD:(0.33,0.33,0.34) ; HR:(0.27,0.73) TMP:( 0.26,0.74) ; DV:(0.77,0.23) VV:(0.28,0.72) ; O3:(0.59,0.41)	(1,12,16,18) 7 etiq.	0.7526	51.4089	0.0169
		2	(9,2,2,7,8,3)	HD:(0.05,0.11,0.18,0.05,0.15,0.05,0.13,0.12,0.16) HR:(0.36,0.64) ; TMP:(0.40,0.60) DV:(0.40,0.14,0.05,0.13,0.12,0.06,0.10) VV:(0.12,0.12,0.12,0.12,0.13,0.13,0.13,0.13) O3:(0.29,0.33,0.38)	(7,12,18) 12 etiq.	0.609	85.7378	0.0192
3		(2,3,2,2,2,3)	HD:( 0.67,0.33) ; HR:( 0.33,0.33,0.34) TMP:( 0.38,0.62) ; DV:( 0.66,0.34) VV:( 0.41,0.59) O3:( 0.41,0.41,0.18)	(7,10,12,16,18) 7 etiq.	0.629	48.712	0.0208	
4		(5,3,7,8,4,2)	HD:(0.05,0.26,0.05,0.19,0.45) ; HR:(0.28,0.53,0.19) TMP:( 0.26,0.10,0.22,0.19,0.07,0.09,0.07) ; DV:(0.07,0.12,0.16,0.21,0.05,0.18,0.12,0.09) VV:(0.15,0.19,0.33,0.33) ; O3:(0.60,0.40)	(1,12,18) 7 etiq.	0.7397	43.1676	0.0221	

**Tabla 4.** Resultados de la granularidad y las funciones de pertenencia para la estimación de la concentración de ozono en México usando el AG multiobjetivo. Criterio de parada = 8000 evaluaciones. Mes Enero.

# ejec.	Gran.	Proporción de datos	Opt. Mask / $\Sigma$ etiquetas	Q	MSE <sub>entr</sub>	RMS <sub>prue</sub>
A	1	(4,6,8,2,3,3) HD:(0.28,0.35, 0.2,0.17) HR:(0.16,0.16,0.17,0.17,0.17,0.17) TMP:(0.12,0.12,0.12,0.12,0.13,0.13,0.13,0.13) DV:(0.67,0.33) ; VV:(0.33,0.33,0.34) O3:(0.41,0.20,0.39)	(1,4,12,16,18) 9 etiq.	0.6677	43.834	0.0190
		(3,4,3,4,2,2) HD:(0.33,0.33,0.34) ; HR:(0.25,0.25,0.25,0.25) TMP:(0.33,0.33,0.34) ; DV:(0.25,0.25,0.25,0.25) VV:(0.11,0.89) ; O3:(0.83,0.17)	(12,18) 2 etiq.	0.7522	106.9838	0.0363
	2	(2,3,2,2,2,2) HD:(0.75,0.25) ; HR:(0.33,0.33,0.34) TMP:(0.55,0.45) ; DV:(0.78,0.22) VV:(0.20,0.80) ; O3:(0.68,0.32)	(4,12,13,16,18) 6 etiq.	0.7486	42.007	0.0191
		(3,4,2,2,2,2) HD:(0.33,0.33,0.34) ; HR:(0.20,0.20,0.18,0.42) TMP:(0.36,0.64) ; DV:(0.63,0.37) VV:(0.52,0.48) ; O3:(0.84,0.16)	(12,18) 2 etiq.	0.756	106.2713	0.0361
	3	(4,3,2,3,3,2) HD:(0.25,0.25,0.25,0.25) ; HR:(0.33,0.33,0.34) TMP:(0.13,0.87) ; DV:(0.46,0.48,0.06) VV:(0.33,0.33,0.34) ; O3:(0.78,0.22)	(8,13,17,18) 12 etiq.	0.7562	42.8306	0.0169
		(3,2,3,2,3,2) HD:(0.19,0.17,0.64) ; HR:(0.76,0.24) TMP:(0.33,0.33,0.34) ; DV:(0.53,0.47) VV:(0.43,0.43,0.14) ; O3:(0.78,0.22)	(12,18) 2 etiq.	0.7254	112.7337	0.0378
	4	(5,4,3,6,5,3) HD:(0.15,0.15,0.15,0.15,0.15,0.40) HR:(0.14,0.13,0.13,0.60) ; TMP:(0.25,0.25,0.50) DV:(0.16,0.16,0.17,0.17,0.17,0.17) VV:(0.25,0.18,0.19,0.19,0.19) ; O3:(0.33,0.33,0.34)	(1,12,18) 8 etiq.	0.6307	43.0066	0.0215
		(3,3,3,2,2,2) HD:(0.18,0.65,0.17) ; HR:(0.33,0.33,0.34) TMP:(0.33,0.33,0.34) ; DV:(0.63,0.37) VV:(0.94,0.06) ; O3:(0.84,0.16)	(12,18) 2 etiq.	0.756	106.2713	0.0361
B	1	(4,4,8,2,3,3) HD:(0.28,0.35, 0.2,0.17) ; HR:(0.22,0.22,0.23,0.33) TMP:(0.13,0.13,0.13,0.13,0.14,0.14,0.07,0.13) ; DV:(0.67,0.33) ; VV:(0.17,0.63,0.20) O3:(0.17,0.63,0.20)	(1,12,16,18) 9 etiq.	0.651	54.0317	0.0180
	2	(2,3,2,2,2,2) HD:(0.75,0.25) ; HR:(0.33,0.33,0.34) TMP:(0.55,0.45) ; DV:(0.78,0.22) VV:(0.20,0.80) ; O3:(0.68,0.32)	(4,12,13,16,18) 6 etiq.	0.7486	42.007	0.0191
	3	(4,3,2,3,3,2) HD:(0.25,0.25,0.25,0.25) ; HR:(0.33,0.33,0.34) TMP:(0.13,0.87) ; DV:(0.46,0.48,0.06) VV:(0.33,0.33,0.34) ; O3:(0.78,0.22)	(8,13,17,18) 12 etiq.	0.7562	42.8306	0.0169
	4	(4,2,2,2,3,2) HD:(0.25,0.25,0.25,0.25) ; HR:(0.64,0.36) TMP:(0.54,0.46) ; DV:(0.52,0.48) VV:(0.33,0.33,0.34) ; O3:(0.84,0.16)	(4,11,13,16,18) 11 etiq.	0.7625	55.0864	0.0184

## 5.2 Análisis de resultados y conclusiones

Este bloque comprende un análisis de los resultados obtenidos por el AG multiobjetivo propuesto en esta investigación para la resolución del problema de estimación de concentraciones de ozono en la zona centro del Área Metropolitana de la Ciudad de México. Para finalizar se proporcionan algunas conclusiones.

De las *Tablas 2, 3, 4 y 5* se puede indicar que el AG multiobjetivo cumplió su tarea de encontrar un conjunto de soluciones no dominadas que satisfacen el propósito de minimizar los valores de  $\Sigma$ etiq y MSE<sub>entr</sub>. En la sección *A* de cada tabla se observan un conjunto de soluciones no dominadas por cada ejecución que

**Tabla 5.** Resultados de la granularidad y las funciones de pertenencia para la estimación de la concentración de ozono en México usando el AG multiobjetivo. Criterio de parada = 16000 evaluaciones. Mes Enero.

# ejec.	Gran.	Proporción de datos	Opt. Mask / $\Sigma$ etiquetas	Q	MSE <sub>entr</sub>	RMS <sub>prue</sub>
A	1	(4,3,4,4,3,2) HD:(0.25,0.25,0.25,0.25); HR:(0.38,0.21,0.41) TMP:(0.14, 0.6,0.13,0.13) DV:(0.25,0.25,0.25,0.25) VV:(0.49,0.14,0.37); O3:(0.84,0.16)	(8,13,17,18) 12 etiq.	0.7708	43.551	0.0172
		(3,3,2,2,2,2) HD:(0.20,0.58,0.22); HR:(0.28,0.45,0.27) TMP:(0.64,0.36); DV:(0.50,0.50) VV:(0.35,0.65); O3:(0.84,0.16)	(12,18) 2 etiq.	0.756	106.2713	0.0361
	2	(2,4,5,3,3,3) HD:(0.77,0.23); HR:(0.54,0.16,0.15,0.15) TMP:(0.48,0.12,0.13,0.13,0.14) DV:(0.13,0.73,0.14) VV:(0.46,0.49,0.05); O3:(0.82,0.13,0.05)	(12,13,18) 5 etiq.	0.7873	46.5797	0.0214
		(4,2,3,5,2,2) HD:(0.20,0.22,0.22,0.36); HR:(0.36,0.64) TMP:(0.29,0.17,0.54) DV:(0.20,0.20,0.20,0.20,0.20) VV:(0.50,0.50); O3:(0.80,0.20)	(6,12,18) 2 etiq.	0.7359	111.5277	0.0373
	3	(6,6,2,4,7,2) HD:(0.05,0.05,0.44,0.09,0.16,0.21) HR:(0.26,0.43,0.15,0.06,0.05,0.05); TMP:(0.48,0.52); DV:(0.25,0.25,0.25,0.25) VV:(0.21,0.09,0.24,0.08,0.08,0.09,0.21) O3:(0.42,0.58)	(3,12,15,16,18) 8 etiq.	0.5991	35.1356	0.0222
		(3,2,2,2,3,2) HD:(0.33,0.33,0.34); HR:(0.50,0.50) TMP:(0.50,0.50); DV:(0.50,0.50) VV:(0.33,0.33,0.34) O3:(0.80,0.20)	(6,12,18) 2 etiq.	0.7359	111.5277	0.0373
	4	(3,4,2,2,2,2) HD:(0.21,0.67,0.12); HR:(0.25,0.25,0.25,0.25) TMP:(0.86,0.14); DV:(0.79,0.21) VV:(0.16,0.84); O3:(0.84,0.16)	(3,4,12,15,18) 6 etiq.	0.7566	39.4247	0.0219
		(3,3,3,3,2,2) HD:(0.21,0.67,0.12); HR:(0.39,0.30,0.31) TMP:(0.33,0.33,0.34) DV:(0.33,0.33,0.34) VV:(0.50,0.50) O3:(0.84,0.16)	(12,18) 2 etiq.	0.756	106.2713	0.0361
B	1	(4,3,4,4,3,2) HD:(0.25,0.25,0.25,0.25); HR:(0.38,0.21,0.41) TMP:(0.14,0.60,0.13,0.13) DV:(0.25,0.25,0.25,0.25) VV:(0.49,0.14,0.37); O3:(0.84,0.16)	(8,13,17,18) 12 etiq.	0.7708	43.551	0.0172
	2	(4,5,5,2,3,4) HD:(0.32,0.07,0.31,0.30); HR:(0.20,0.20,0.20,0.20,0.20) TMP:(0.20,0.20,0.20,0.20,0.20); DV:(0.66,0.34) VV:(0.32,0.31,0.37); O3:(0.25,0.25,0.25,0.25)	(12,13,17,18) 11 etiq.	0.5501	65.6164	0.0209
	3	(4,2,3,2,3,2) HD:(0.25,0.25,0.25,0.25); HR:(0.50,0.50) TMP:(0.09,0.08,0.83); DV:(0.89,0.11) VV:(0.33,0.33,0.34); O3:(0.80,0.20)	(8,13,17,18) 11 etiq.	0.7615	40.1446	0.0174
	4	(4,4,2,3,3,2) HD:(0.25,0.25,0.25,0.25) HR:(0.25,0.25,0.25,0.25); TMP:(0.25,0.75) DV:(0.40,0.30,0.30) VV:(0.33,0.33,0.34); O3:(0.84,0.16)	(13,16,17,18) 12 etiq.	0.7636	57.254	0.0164

tienen el menor error MSE<sub>entr</sub> y la menor sumatoria de etiquetas.

En cuanto a la precisión de los modelos (última columna de cada tabla), se observa que las soluciones que poseen la menor sumatoria de etiquetas no poseen un buen RMS<sub>prue</sub>. Las soluciones más simples sugeridas por el AG multiobjetivo no son capaces de predecir correctamente el comportamiento del ozono. Analicemos este hecho con más atención. En la *Tabla 2* vemos que las soluciones con menor  $\Sigma$ etiq en las 4 ejecuciones realizadas, tienen todas ellas un valor de  $\Sigma$ etiq de 2 y la máscara óptima que se encuentra con esta partición es (12,18), es decir que el ozono en el instante actual (posición 18 de la máscara) tiene una relación causal únicamente con la misma variable ozono una hora antes (posición 12 de la máscara). Es evidente que una máscara de complejidad 2 donde la única m-entrada corresponde a la variable de salida en el instante de tiempo anterior



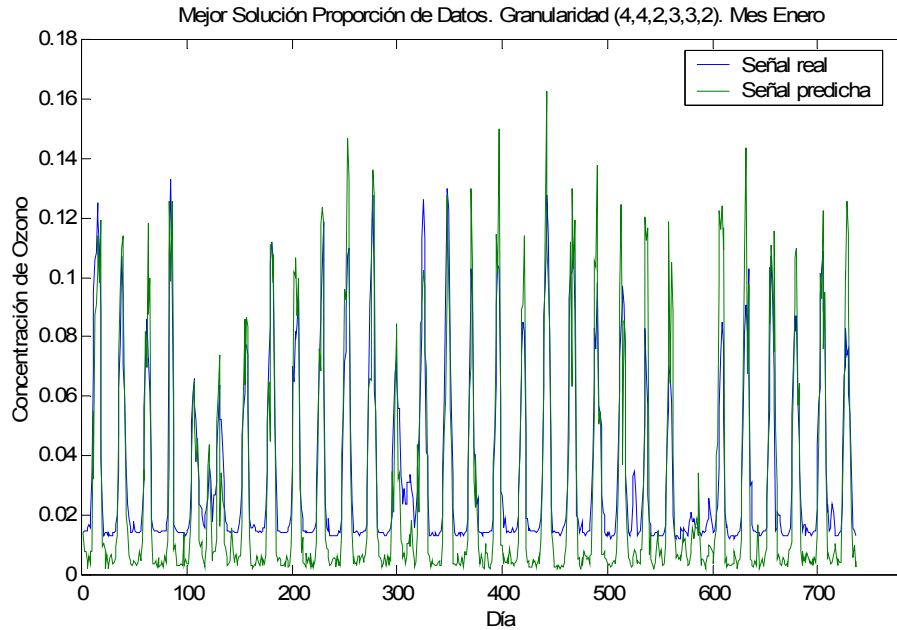
**Tabla 6.** Tiempo consumido, número de generaciones y número de soluciones no dominadas obtenidas por ejecución y criterio de parada del AG multiobjetivo. Caso Estudio: Ozono México, mes Enero.

Núm. evaluaciones: 2000				
Tiempo Promedio	# ejec.	# Gen.	Tiempo (horas)	# Soluciones no dominadas
10h:11min	1	30	11:18	24
	2	30	12:01	25
	3	30	7:52	39
	4	30	9:31	23
Núm. evaluaciones: 4000				
Tiempo Promedio	# ejec.	# Gen.	Tiempo (horas)	# Soluciones no dominadas
18h:17min	1	61	10:40	28
	2	62	19:16	13
	3	62	19:14	17
	4	61	23:57	16
Núm. evaluaciones: 8000				
Tiempo Promedio	# ejec.	# Gen.	Tiempo (horas)	# Soluciones no dominadas
29h:13min	1	124	25:14	18
	2	125	28:41	17
	3	124	35:16	16
	4	124	27:39	11
Núm. evaluaciones: 16000				
Tiempo Promedio	# ejec.	# Gen.	Tiempo (horas)	# Soluciones no dominadas
50h:41min	1	250	58:35	13
	2	250	51:52	16
	3	249	52:22	19
	4	250	39:56	22

tiene una capacidad pésima de predicción, puesto que la señal de predicción será una constante. Recordemos que nuestro objetivo, en la aplicación que nos ocupa, es realizar una predicción a largo plazo. También aparece en las tablas, en varias ocasiones, la máscara (6,12,18), donde la posición 6 corresponde, una vez más, a un valor anterior de la variable ozono. Por lo tanto, nos encontramos en la misma situación que teníamos con la máscara (12,18). Por este motivo, estas máscaras deben ser rechazadas como posibles soluciones. Este tipo de máscaras aparecen también en las *Tablas 3, 4 y 5*, pues lógicamente el **AGMO3** las encuentra como mejores soluciones desde la perspectiva de minimizar el objetivo  $\Sigma_{eti}$ . En este sentido, se podría pensar en incluir en el **AGMO3** una penalización a la obtención de este tipo de máscara. Esto se lograría de manera simple forzando que el mínimo valor de  $\Sigma_{eti}$  fuera mayor a la granularidad más pequeña de las variables que intervienen en el sistema.

Si analizamos los valores de  $RMS_{prue}$  en aquellas soluciones (sección *A* de las tablas) donde se está minimizando el objetivo  $MSE_{entr}$ , vemos que las soluciones con mayor error de prueba (0.0375 y 0.0222) son aquellas en las que la máscara derivada de dicha solución no incluye la variable hora del día (HD) entre sus relaciones causales con la salida. Esta es la causa fundamental del bajo

rendimiento de estas soluciones en la predicción (*Tabla 3*). Las máscaras que no incluyen alguna de las posiciones 1, 7 o 13, correspondientes a diferentes instantes de tiempo de HD, son las que tienen  $RMS_{prue}$  más altos. Como ya se mencionó en [ANF06b], en esta aplicación la variable HD está realizando el papel de los precursores del ozono, por lo que es fundamental que esta variable forme parte del modelo.



**Figura 2.** Predicción del mejor modelo FIR para el mes Enero. Problema concentraciones de ozono en México. Granularidad (4,4,2,3,3,2) con  $Q=0.7636$  y distribución de landmarks: HD:(0.25,0.25,0.25,0.25)-HR:(0.25,0.25,0.25,0.25)-TMP:(0.25, 0.75)-DV:(0.40,0.30,0.30)-VV:(0.33,0.33,0.34)-O<sub>3</sub>:(0.84,0.16). Máscara:(13,16,17,18).  $RMS_{prue}=0.0164$ . Método **AGMO3**.

El mejor resultado logrado por **FIR EXPERTOS** [NM06] tiene un  $RMS_{prue}$  de 0.0176 ( $\Sigma etiq=10$ ) superior al error de 0.0164 ( $\Sigma etiq=12$ ) obtenido por el **AGMO3**. Asimismo, este método supera los resultados obtenidos en [ANF06b] por **FIR DEFECTO**, **AG1+EFP**, **AG1+AG2** y **AG3**.

El AG multiobjetivo ha permitido en una sola ejecución obtener un conjunto de soluciones (unas más simples y otras más precisas) en un tiempo razonable (*Tabla 6*) al alcanzar el criterio de parada estipulado. De este modo, no es necesario ejecutarlo tantas veces como los métodos **AG1**, **AG2** y **AG3** descritos en trabajos previos.

En resumen, el AG multiobjetivo propuesto en esta investigación ha permitido obtener modelos FIR para el mes de Enero con igual habilidad de

predicción que los obtenidos por [NM06] y **FIR DEFECTO**. La combinación **EXPERTOS + AG2** sigue siendo el método más eficaz para obtener los mejores resultados de predicción del conjunto de datos de prueba, aunque le siguen muy de cerca **AGMO3**, **AG1+AG2** y **AG3**.

La *figura 2* muestra la señal real y de predicción usando el mejor modelo obtenido con el AG multiobjetivo. Analizando la figura vemos que los picos altos de la señal se alcanzan en la mayoría de las ocasiones, aunque hay picos que no se llegan a predecir correctamente.

En conclusión, se puede indicar que el AG multiobjetivo propuesto es una primera aproximación al desarrollo de este tipo de AGs para la determinación de los parámetros de fusificación de la metodología FIR. Consideramos que es necesario probarlo en un conjunto más amplio de aplicaciones para conocer bien su comportamiento. Así mismo, sería muy interesante desarrollar un AGMO de segunda generación para obtener mejores conjuntos de soluciones (más cercanos al *Pareto*).

## Referencias Bibliográficas

- [AN05] J. Acosta and A. Nebot, "Uso del razonamiento inductivo difuso para solucionar un problema de distribución de energía eléctrica," Report de Recerca LSI-05-22-R, Dept. Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, Barcelona, Spain, 2005.
- [ANF04] J. Acosta, A. Nebot and J. M. Fuertes, "Un método de aprendizaje de las funciones de pertenencia para razonamiento inductivo," Report de Recerca ESAII-RR-04-03, Dept. d'Enginyeria de Sistemes, Automàtica i Informàtica Industrial, Universitat Politècnica de Catalunya, Barcelona, Spain, 2004.
- [ANF05a] J. Acosta, A. Nebot and J. M. Fuertes, "Modelado del Sistema Nervioso Central Humano usando una metodología híbrida de Soft Computing," Report de Recerca LSI-05-57-R, Dept. Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, Barcelona, Spain, 2005.
- [ANF05b] J. Acosta, A. Nebot and J. M. Fuertes, "Aprendizaje Global de los parámetros de discretización del Razonamiento Inductivo Difuso," Report de Recerca ESAII-RR-05-10, Dept. d'Enginyeria de Sistemes, Automàtica i Informàtica Industrial, Universitat Politècnica de Catalunya, Barcelona, Spain, 2005.
- [ANF06a] J. Acosta, A. Nebot and J. M. Fuertes, "Predicción de concentraciones locales de ozono usando una metodología híbrida

- de Soft Computing,” Report de Recerca ESAII-RR-06-06, Dept. d’Enginyeria de Sistemes, Automàtica i Informàtica Industrial, Universitat Politècnica de Catalunya, Barcelona, Spain, 2006.
- [ANF06b] J. Acosta, A. Nebot and J. M. Fuertes, “Modelado de las concentraciones locales de ozono en la Zona Centro del Area Metropolitana de la Ciudad de México,” Report de Recerca LSI-06-34-R, Dept. Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, Barcelona, Spain, 2006.
- [Bak87] J. E. Baker, “Reducing bias and inefficiency in the selection algorithm,” in *Proc. 2nd Int. Conf. Genetic Algorithms (ICGA’87)*, Hillsdale, 1987, pp. 14-21.
- [BBTZ01] S. Bleuler, M. Braek, L. Thiele, and E. Zitzler, “Multiobjective Genetic Programming: Reducing Bloat Using SPEA2,” in *Proc. IEEE 2001 Congress on Evolutionary Computation (CEC’2001)-Vol. 1*, New Jersey, 2001, pp. 536-543.
- [BSEK06] P.P. Bonissone, R. Subbu, N. Eklund, and T.R. Kiehl, “Evolutionary Algorithms + Domain Knowledge = Real-World Evolutionary Computation”, *IEEE Trans. Evol Comput.*, vol. 10, no. 3, pp. 256-280, June 2006.
- [BW01] A.L. Blumel and B.A. White, “Multiobjective Optimization of Fuzzy Logic Scheduled Controllers for Missile Autopilot Design,” in *IEEE Joint 9th IFSA World Congress and 20th NAFIPS Int. Conf.- Vol. 3*, 2001, pp. 1758-1763.
- [CCA03] H.-S. Chung, S. Choi, and J.J. Alonso, “Supersonic Business Jet Design Using Knowledge-Based Genetic Algorithm with Adaptive Unstructured Grid Methodology,” presented at the 21st AIAA Applied Aerodynamics Conference, Orlando, FL, June 23-26, 2003, Paper AIAA-2003-3791.
- [CHB00] C.A. Coello, A. Hernández, and B.P. Buckles, “Evolutionary Multiobjective Design of Combinational Logic Circuits,” in *Proceedings of the Second NASA/DoD Workshop on Evolvable Hardware*, J. Lohn, A. Stoica, D. Keymeulen, and S. Colombano, Eds. IEEE Computer Society, 2000, pp. 161-170.
- [CJKO01] D.W. Corne, N.R. Jerram, J.D. Knowles and M.J. Oates, “PESA-II: Region-based Selection in Evolutionary Multiobjective Optimization,” in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, L. Spector, E. Goodman, A. Wu, W.B. Langdon, H.-M. Voigt, M. Gen, S. Sen, M. Dorigo, S. Pezeshk, M.H. Garzon and E. Burke, Eds. Morgan Kaufmann Publishers, 2001, pp. 283-290.
- [CKO00] D.W. Corne, J.D. Knowles and M.J. Oates, “The Pareto Envelope-based Selection Algorithm for Multiobjective Optimization,” in

- Proceedings of the Parallel Problem Solving from Nature VI Conference*, M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J.J. Merelo and H.-P. Schwefel, Eds. Springer, 2000, pp. 839-848.
- [Coe00] C.A. Coello, "Constraint-handling using an evolutionary multiobjective optimization technique," *Civil Engineering and Environmental Systems*, vol. 17, pp. 319-346, 2000.
- [Coe03] C.A. Coello, "Evolutionary Multiobjective Optimization: Current and Future Challenges," in *Advances in Soft Computing: Engineering, Design and Manufacturing*, J. Benitez, O. Cordón, F. Hoffmann and R. Roy, Eds. Springer-Verlag, 2003, pp. 243-256.
- [Coe99] C.A. Coello, "A Comprehensive Survey of Evolutionary-Based Multiobjective Optimization Techniques," *Knowledge and Information Systems. An International Journal*, vol. 1, no. 3, pp. 269-308, 1999.
- [CRFC+03] P.B. Cheung, L.F.R. Reis, K.T.M. Formiga, F.H. Chaudhry and W. G.C. Ticona, "Multiobjective Evolutionary Algorithms Applied to the Rehabilitation of a Water Distribution System: A Comparative Study," in *Second International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, C.M. Fonseca, P.J. Fleming, E. Zitzler, K. Deb and L. Thiele, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 2632, 2003, pp. 662-676.
- [CT01a] C.A. Coello and G. Toscano, "Multiobjective Optimization using a Micro-Genetic Algorithm," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, L. Spector, E. Goodman, A. Wu, W.B. Langdon, H.-M. Voigt, M. Gen, S. Sen, M. Dorigo, S. Pezeshk, M.H. Garzon and E. Burke, Eds. Morgan Kaufmann Publishers, 2001, pp. 274-282.
- [CT01a] C.A. Coello and G. Toscano, "Multiobjective Optimization using a Micro-Genetic Algorithm," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, L. Spector, E. Goodman, A. Wu, W.B. Langdon, H.-M. Voigt, M. Gen, S. Sen, M. Dorigo, S. Pezeshk, M.H. Garzon and E. Burke, Eds. Morgan Kaufmann Publishers, 2001, pp. 274-282.
- [CT01b] C.A. Coello and G. Toscano, "A Micro-Genetic Algorithm for Multiobjective Optimization," in *First International Conference on Evolutionary Multi-Criterion Optimization*, E. Zitzler, K. Deb, L. Thiele, C.A. Coello and D. Corne, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 1993, 2001, pp. 126-140.
- [CT01b] C.A. Coello and G. Toscano, "A Micro-Genetic Algorithm for Multiobjective Optimization," in *First International Conference on Evolutionary Multi-Criterion Optimization*, E. Zitzler, K. Deb,

- L. Thiele, C.A. Coello and D. Corne, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 1993, 2001, pp. 126–140.
- [CVL02] C.A. Coello, D.V. Veldhuizen, and G.B. Lamont, *Evolutionary Algorithms for Solving Multi-Objective Problems*. New York: Kluwer Academic/Plenum Publishers, 2002.
- [DAPM00] K. Deb, S. Agrawal, A. Pratab and T. Meyarivan, "A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II," KanGAL report 200001, Indian Institute of Technology, Kanpur, India, 2000.
- [DdWdB01] E.I. Ducheyne, R.R. De Wulf, and B. De Baets, "Bi-objective genetic algorithm for forest management: a comparative study," in *Proc. 2001 Genetic and Evolutionary Computation Conf. – Late Breaking Papers*, California, 2001, pp. 63-66.
- [Deb01] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley and Sons, 2001.
- [Deb91] K. Deb, "Binary and Floating-Point Function Optimzation using Messy Genetic Algorithms," Ph.D. thesis, Dept. Engineering Mechanics, University of Alabama, Tuscaloosa, AL 35487, 1991.
- [DG01] K. Deb and T. Goel, "A Hybrid Multi-Objective Evolutionary Approach to Engineering Shape Design," in *First International Conference on Evolutionary Multi-Criterion Optimization*, E. Zitzler, K. Deb, L. Thiele, C.A. Coello and D. Corne, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 1993, 2001, pp. 385-399.
- [DJ98] R.P. Dick and N.K. Jha, "MOGAC: A Multiobjective Genetic Algorithm for Hardware-Software Co-synthesis of Hierarchical Heterogeneous Distributed Embedded Systems," *IEEE Trans. Computer-Aided Design*, vol. 17, no. 10, pp. 920-935, Oct. 1998.
- [DPAM02] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, pp. 182-197, Apr. 2002.
- [DR03] J.L. Dorn and S.R. Ranjithan, "Evolutionary Multiobjective Optimization in Watershed Water Quality Management," in *Second International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, C.M. Fonseca, P.J. Fleming, E. Zitzler, K. Deb and L. Thiele, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 2632, 2003, pp. 692-706.
- [DRFF00] N.M. Duarte, A. E. Ruano, C.M. Fonseca, and P.J. Fleming, "Accelerating Multi-Objective Control System Design Using a Neuro-Genetic Approach," in *Proc. 2000 IEEE Congress on Evolutionary Computation - Vol. 1*, New Jersey, 2000, pp. 392-

- 397.
- [EHM00] C. Emmanouilidis, A. Hunter, and J. MacIntyre, "A Multiobjective Evolutionary Setting for Feature Selection and a Commonality-Based Crossover Operator," in *Proc. IEEE 2000 Congress on Evolutionary Computation- Vol. 1*, New Jersey, 2000, pp. 309-316.
- [EMH01] M. Erickson, A. Mayer, and J. Horn, "The Niche Pareto Genetic Algorithm 2 Applied to the Design of Groundwater Remediation Systems," in *First International Conference on Evolutionary Multi-Criterion Optimization*, E. Zitzler, K. Deb, L. Thiele, C.A. Coello and D. Corne, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 1993, 2001, pp. 681-695.
- [FDA04] M. Farina, K. Deb and P. Amato, "Dynamic multiobjective optimization problems: test cases, approximations and applications," *IEEE Trans. Evol. Comput.*, vol. 8, no. 5, pp. 182-197, Oct. 2004.
- [FF93] C.M. Fonseca, and P.J. Flemming, "Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization," in *Proceedings of the Fifth International Conference on Genetic Algorithms*, S. Forrest, Ed. Morgan Kaufmann Publishers, 1993, pp. 416-423.
- [FF94] C.M. Fonseca, and P.J. Flemming, "An Overview of Evolutionary Algorithms in Multiobjective Optimization," Technical Report, Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, U.K., 1994.
- [FF95] C.M. Fonseca, and P.J. Flemming, "An Overview of Evolutionary Algorithms in Multiobjective Optimization," *Evol. Comput.*, vol. 3, no. 1, pp. 1-16, 1995.
- [FMSZ03] W. Fornaciari, P. Micheli, F. Salice, and L. Zampella, "A First Step Towards Hw/Sw Partitioning of UML Specifications," in *IEEE/ACM Design Automation and Test in Europe (DATE'03)*, Munich, 2003, pp. 668-673.
- [GD91] D.E. Goldberg and K. Deb, "A comparative analysis of selection schemes used in genetic algorithms," in *Foundations of Genetic Algorithms 1*, G.J.E. Rawlins, Ed. Morgan Kaufmann Publishers, 1991, pp. 69-93.
- [GDKH93] D. E. Goldberg, K. Deb, H. Kargupta, and G. Harik, "Rapid, accurate optimization of difficult problems using fast messy genetic algorithms," in *Proceedings of the Fifth International Conference on Genetic Algorithms*, S. Forrest, Ed. Morgan Kaufmann Publishers, 1993, pp. 56-64.

- [GGW03] D. Greiner, B. Galván, and G. Winter, "Safety Systems Optimum Design by Multicriteria Evolutionary Algorithms," in *Second International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, C.M. Fonseca, P.J. Fleming, E. Zitzler, K. Deb and L. Thiele, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 2632, 2003, pp. 722-736.
- [GLM00] I.E. Golovkin, S.J. Louis, and R.C. Mancini, "Parallel Implementation of Niche Pareto Genetic Algorithm Code for X-ray Plasma Spectroscopy," in *Proc. 2000 Genetic and Evolutionary Computation Conf. – Late Breaking Papers*, Nevada, 2000, pp. 222-227.
- [Gol89] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
- [GPVC02] A. Gaspar-Cunha, A. Poulesquen, B. Vergnes, and J. A. Covas, "Optimization of Processing Conditions for Polymer Twin-Screw Extrusion", *Intern. Polym. Process.*, vol. 17, no. 3, p. 201-213, 2002.
- [GR87] D.E. Goldberg and J. Richardson, "Genetic algorithm with sharing for multimodal function optimization," in *Proc. 2nd Int. Conf. Genetic Algorithms (ICGA'87)*, Hillsdale, 1987, pp. 41-49.
- [Gre84] J.J. Grefenstette, "GENESIS: A system for using genetic search procedures," in *Proc. 1984 Conf. Intell. Systems and Machines*, Rochester, MI, 1984, pp. 161-165.
- [Gre86] J. J. Grefenstette, "Optimization of Control Parameters for Genetic Algorithms," *IEEE Trans. Syst., Man, Cybern.*, vol. 16, no. 1, pp. 122-128, Jan./Feb. 1986.
- [HBP02] A. Herreros, E. Baeyens, and J.R. Perán, "MRCD: a genetic algorithm for multiobjective robust control design," Genetic Algorithm," *Engineering Applications of Artificial Intelligence*, vol. 15, no. 3-4 , pp. 285-301, June-Aug. 2002.
- [HG97] P. Haastrup and Â. Guimarães, "Exploring the use of Multi-Objective Genetic Algorithms for Reducing Traffic Generated Urban Air and Noise Pollution," in *Proc. 5th European Cong. Intelligent and Soft Computing*, Aachen, 1997, pp. 819-825.
- [HL92] P. Hajela and C. Y. Lin, "Genetic search strategies in multicriterion optimal design," *Structural Optimization*, vol. 4, pp. 99-107, 1992.
- [HN93] J. Horn and N. Nafpliotis, "Multiobjective Optimization using the Niche Pareto Genetic Algorithms," IlliGAL Report 93005, Department of Computer Science, University of Illinois, Urbana, Champaign, 1993.



- [KC00] J.D. Knowles and D.W. Corne, "Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy," *Evol. Comput.*, vol. 8, no. 2, pp. 149-172, 2000.
- [KC99a] J.D. Knowles and D.W. Corne, "The Pareto Archived Evolution Strategy: A New Baseline Algorithm for Multiobjective Optimisation," in *Proc. 1999 Cong. Evolut. Comput. (CEC'99)*, Washington D.C., USA, July 6-9, 1999, pp. 98-105.
- [KC99a] J.D. Knowles and D.W. Corne, "The Pareto Archived Evolution Strategy: A New Baseline Algorithm for Multiobjective Optimisation," in *Proc. 1999 Cong. Evolut. Comput. (CEC'99)*, Washington D.C., USA, July 6-9, 1999, pp. 98-105.
- [KC99b] J.D. Knowles and D.W. Corne, "Assessing the Performance of the Pareto Archived Evolution Strategy," in *Proceedings of the 1999 Genetic and Evolutionary Computation Conference. Workshop Program*, Orlando, FL, 1999, pp. 123-124.
- [KCO00] J.D. Knowles, D.W. Corne, and M.J. Oates, "On the Assessment of Multiobjective Approaches to the Adaptive Distributed Database Management Problem," in *Proceedings of the Sixth International Conference on Parallel Problem Solving from Nature (PPSN VI)*, M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J.J. Merelo and H.-P. Schwefel, Eds. Springer, 2000, pp. 869-878.
- [LE97] J. Lis and A.E. Eiben, "A Multi-Sexual Genetic Algorithm for Multiobjective Optimization," in *Proc. 1997 IEEE Int. Conf. Evolut. Comput. (ICEC'97)*, Indianapolis, USA, April 13-16, 1997, pp. 59-64.
- [LMBZ01] M. Lahanas, N. Milickovic, D. Baltas, and N. Zamboglou, "Application of Multiobjective Evolutionary Algorithms for Dose Optimization Problems in Brachytherapy," in *First International Conference on Evolutionary Multi-Criterion Optimization*, E. Zitzler, K. Deb, L. Thiele, C.A. Coello and D. Corne, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 1993, 2001, pp. 574-587.
- [LSMB03] M. Lahanas, E. Schreibmann, N. Milickovic, and D. Baltas, "Intensity Modulated Beam Radiation Therapy Dose Optimization with Multiobjective Evolutionary Algorithms," in *Second International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, C.M. Fonseca, P.J. Fleming, E. Zitzler, K. Deb and L. Thiele, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 2632, 2003, pp. 648-661.
- [Mar97] T. Marcu, "A multiobjective evolutionary approach to pattern recognition for robust diagnosis of process faults," in *Proc. IFAC Symposium on Fault Detection, Supervision and Safety for*

- Technical Processes: SAFEPROCESS'97*, Hull, UK, 1997, pp. 1183-1188.
- [MCH98] W. Mason, V. Coverstone-Carroll, and J. Hartmann, "Optimal Earth Orbiting Satellite Constellations via a Pareto Genetic Algorithm," in *1998 AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, Massachusetts, pp. 169-177, 1998.
- [MI95] T. Murata and H. Ishibuchi, "MOGA: Multi-objective genetic algorithms," in *Proc. 2nd IEEE Int. Conf. Evol. Comput.*, Perth, Australia, November 29-December 1, 1995, pp. 289-294.
- [Mic96] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*. New York: Springer-Verlag, 1996.
- [MLDP99] N. Marco, S. Lanteri, J.-A. Desideri, and J. Périaux, "A Parallel Genetic Algorithm for Multi-Objective Optimization in Computational Fluid Dynamics," in *Evolutionary Algorithms in Engineering and Computer Science*, K. Miettinen, M.M. Mäkelä, P. Neittaanmäki, and J. Périaux, Eds. John Wiley & Sons, 1999, pp. 445-456.
- [Mor80] J. N. Morse, "Reducing the size of the nondominated set: Pruning by clustering," *Computers and Operations Research*, vol. 7, no. 1-2, pp. 55-66, 1980.
- [NA99] S. Narayanan and S. Azarm, "On Improving Multiobjective Genetic Algorithms for Design Optimization," *Structural Optimization*, vol. 18, pp. 146-155, 1999.
- [NM06] A. Nebot and V. Mugica, "Finding qualitative patterns in ozono behavior," in *Proceedings of 5<sup>th</sup> Mexican International Conference on Artificial Intelligence (MICAI'06)*, Apizaco, México, 2006, pp. 91-100.
- [OGC91] C. K. Oei, D. E. Goldberg, and S.-J. Chang. Tournament Selection, Niching, and the Preservation of Diversity. Technical Report 91011, University of Illinois at Urbana-Champaign, Urbana, Illinois, USA, 1991.
- [PM01] A. Petrovski and J. McCall, "Multi-objective Optimisation of Cancer Chemotherapy Using Evolutionary Algorithms," in *First International Conference on Evolutionary Multi-Criterion Optimization*, E. Zitzler, K. Deb, L. Thiele, C.A. Coello, and D. Corne, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 1993, 2001 pp. 531-545.
- [RER94] B.J. Ritzel, J.W. Eheart, and S. Ranjithan, "Using genetic algorithms to solve a multiple objective groundwater pollution containment problem," *Water Resources Research*, vol. 30, no. 5, pp. 1589-1603, May 1994.

- [RMG01] Patrick M. Reed, Barbara S. Minsker and David E. Goldberg, "A multiobjective approach to cost effective long-term groundwater monitoring using an elitist nondominated sorted genetic algorithm with historical data," *Journal of Hydroinformatics*, vol. 3, no. 2, pp. 71-89, April 2001.
- [Rog00] J.L. Rogers, "A Parallel Approach to Optimum Actuator Selection With A Genetic Algorithm," presented at the AIAA Guidance, Navigation, and Control Conf. *and Exhibit*, Denver, CO, Aug. 14-17, 2000, Paper AIAA-2000-4484.
- [Ros67] R. S. Rosenberg, "Simulation of genetic populations with biochemical properties," Ph.D. thesis, University of Michigan, Ann Harbor, Michigan, 1967.
- [RPLH89] J.T. Richardson, M. R. Palmer, G. Liepins, and M. Hilliard, "Some Guidelines for Genetic Algorithms with Penalty Functions," in *Proceedings of the Third International Conference on Genetic Algorithms (ICGA-89)*, J. D. Schaffer, Ed. Morgan Kaufmann Publishers, 1989, pp. 191-197.
- [Sch84] J.D. Schaffer, "Some experiments in machine learning using vector evaluated genetic algorithms," Unpublished doctoral dissertation, Vanderbilt University, 1984.
- [Sch85] J.D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," in *Proc. 1st Int. Conf. Genetic Algorithms and Their Applications*, Pittsburgh, PA, July 24-26, 1985, pp. 93-100.
- [Sch95b] J.R. Schott, "Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization," Master's thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1995.
- [SD93] N. Srinivas and K. Deb, "Multiobjective Optimization using Nondominated Sorting in Genetic Algorithms," Technical Report, Department of Mechanical Engineering, Indian Institute of Technology, Kanpur, India, 1993.
- [SD94] N. Srinivas and K. Deb, "Multiobjective Optimization using Nondominated Sorting in Genetic Algorithms," *Evol. Comput.*, vol. 2, no. 3, pp. 221-248, 1994.
- [SR97] P.D. Surry and N. J. Radcliffe, "The COMOGA Method: Constrained Optimisation by Multiobjective Genetic Algorithms," *Control and Cybernetics*, vol. 26, no. 3, pp. 391-412, 1997.
- [TC03] G. Toscano and C.A. Coello, "The Micro Genetic Algorithm 2: Towards Online Adaptation in Evolutionary Multiobjective Optimization," in *Second International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, C.M.

- Fonseca, P.J. Fleming, E. Zitzler, K. Deb and L. Thiele, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 2632, 2003, pp. 252-266.
- [TC03] G. Toscano and C.A. Coello, "The Micro Genetic Algorithm 2: Towards Online Adaptation in Evolutionary Multiobjective Optimization," in *Second International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, C.M. Fonseca, P.J. Fleming, E. Zitzler, K. Deb and L. Thiele, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 2632, 2003, pp. 252-266.
- [Thr91] P. Thrift, "Fuzzy logic synthesis with genetic algorithms," in *Proc. 4th Int. Conf. Genetic Algorithms (ICGA'91)*, San Diego, CA, 1991, pp. 509-513.
- [TIG98] T. Murata, H. Ishibuchi, and M. Gen, "Random Weights in Multi-Objective Genetic Algorithms," in *Proc. 2nd Int. Conf. on Engineering Design and Automation*, Maui, Hawaii, 1998. Only CD-ROM Proceedings available.
- [Tos05] G. Toscano, "On the Use of Self-Adaptation and Elitism for Multiobjective Particle Swarm Optimization," Ph.D. thesis, Computer Science Section, Department of Electrical Engineering, CINVESTAV-IPN, September 2005.
- [VL00a] D.A. Van Veldhuizen and G.B. Lamont, "Multiobjective evolutionary algorithms: Analyzing the state-of-the-art," *Evol. Comput.*, vol. 8, no. 2, pp. 125-147, 2000.
- [VL00a] D.A. Van Veldhuizen and G.B. Lamont, "Multiobjective evolutionary algorithms: Analyzing the state-of-the-art," *Evol. Comput.*, vol. 8, no. 2, pp. 125-147, 2000.
- [VL00b] D.A. Van Veldhuizen and G.B. Lamont, "Multiobjective Optimization with Messy Genetic Algorithms," in *Proc. 2000 ACM Symposium on Applied Computing*, Villa Olmo, Italy, 2000, pp. 470-476.
- [WB03] L. Willmes and T. Bäck, "Multi-criteria Airfoil Design with Evolution Strategies," in *Second International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, C.M. Fonseca, P.J. Fleming, E. Zitzler, K. Deb and L. Thiele, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 2632, 2003, pp. 782-795.
- [ZDT00] E. Zitzler, K. Deb and L. Thiele, "Comparison of Multiobjective Evolutionary Algorithms: Empirical Results," *Evol. Comput.*, vol. 8, no. 2, pp. 173-195, 2000.
- [ZLT01] E. Zitzler, M. Laumanns and L. Thiele, "SPEA2: Improving the Strength Pareto Evolutionary Algorithm," Technical Report 103,

- Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, 2001.
- [ZLV00] J.B. Zydallis, G.B. Lamont and D.A. Van Veldhuizen, "Messy Genetic Algorithm Based Multi-Objective Optimization: A Comparative Statistical Analysis," in *PPSN/SAB Workshop on Multiobjective Problem Solving from Nature (MPSN)*, Paris, France, 2000, pp. 1-3. Available: <http://neo.lcc.uma.es/emoo/EMOOconferences.html>
- [ZT98] E. Zitzler and L. Thiele, "An Evolutionary Algorithm for Multiobjective Optimization: The Strength Pareto Approach," Technical Report 43, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, 1998.
- [ZT99] E. Zitzler and L. Thiele, "Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257-271, Nov. 1999.
- [ZT99] E. Zitzler and L. Thiele, "Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257-271, Nov. 1999.
- [ZTB99] E. Zitzler, J. Teich, and S.S. Bhattacharyya, "Evolutionary Algorithm Based Exploration of Software Schedules for Digital Signal Processors," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'99)-Vol. 2*, W. Banzhaf, J. Daida, A. E. Eiben, M. H. Garzon, V. Honavar, M. Jakiela, and R. E. Smith, Eds. Morgan Kaufmann, 1999, pp. 1762-1769.
- [ZVL01] J.B. Zydallis, D.A. Van Veldhuizen and G.B. Lamont, "A Statistical Comparison of Multiobjective Evolutionary Algorithms Including the MOMGA-II," in *First International Conference on Evolutionary Multi-Criterion Optimization*, E. Zitzler, K. Deb, L. Thiele, C.A. Coello and D. Corne, Eds. Springer-Verlag, Lecture Notes in Computer Science No. 1993, 2001, pp. 226-240.

